Provably Invariant Learning without Domain Information

Xiaoyu Tan [*1] **Yong Lin** [*2] **Shengyu Zhu** [3] **Chao Qu** [1] **Xihe Qiu** [4] **Yinghui Xu** [5] **Peng Cui** [6] **Yuan Qi** [5]

Abstract

Typical machine learning applications always assume the data follows independent and identically distributed (IID) assumptions. In contrast, this assumption is frequently violated in real-world circumstances, leading to the Out-of-Distribution (OOD) generalization problem and a major drop in model robustness. To mitigate this issue, the invariance learning technique is leveraged to distinguish between spurious features and invariant features among all input features and to train the model purely on the basis of the invariant features. Numerous invariance learning strategies imply that the training data should contain domain information. Such information includes the environment index or auxiliary information acquired from prior knowledge. However, acquiring these information is typically impossible in practice. In this study, we present TIVA for environment-independent invariance learning, which requires no environment-specific information in training data. We discover and prove that, in causal graph, given mild conditions, it is possible to train an environment partitioning policy based on attributes that are independent of the targets and then conduct invariant risk minimization. We examine our method in comparison to other baseline methods, which demonstrate superior performance and excellent robustness under OOD, using multiple benchmarks.

[*]Equal contribution [1]INF Technology (Shanghai) Co., Ltd. Shanghai, China [2]Hong Kong University of Science and Technology, Hong Kong, China [3]Ubiquant Investment, Beijing, China [4]School of Electronic and Electrical Engineering, Shanghai University of Engineering Science, Shanghai, China [5]Artificial Intelligence Innovation and Incubation (AI[3]) Institute, Fudan University, Shanghai, China [6]Department of Computer Science and Technology, Tsinghua University, Beijing, China. Correspondence to: Yong Lin <ylindf@connect.ust.hk>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

1. Introduction

Machine learning based on deep neural networks (DNN), which has been widely used for computer vision, natural language processing, speech recognition, and other applications, has experienced exceptional success in recent decades (LeCun et al., 2015; Gu et al., 2018; He et al., 2016; Chowdhary, 2020; Vaswani et al., 2017; Devlin et al., 2018; Arulkumaran et al., 2017). The training and testing sets of data for machine learning models are often assumed in the same distribution according to the empirical risk minimization (ERM) technique and under the independent identically distributed (IID) assumption. However, the IID assumption can be easily violated in real-world applications. Recent research indicates that the Out-of-Distribution (OOD) generalization problem might lead ERM-trained models to fail catastrophically, especially when the testing distribution deviates significantly from the training distribution (Hendrycks and Gimpel, 2016; Liang et al., 2017; Wang and Deng, 2018). Based on recent literature (Arjovsky et al., 2019; Ben-Tal et al., 2013; Huang et al., 2020; Duchi and Namkoong, 2021), there are several cases to indicate this issue. One concise example is that the ERM-trained model may rely on background information to perform object classification. However, when the background changes (i.e., OOD) but the object remains unchanged, the classifier's performance decreases dramatically (Arjovsky et al., 2019). A popular line of work attempts to extract invariant features $\boldsymbol{X}_v$ which have a stable correlation with target Y and can predict Y stably in the novel testing distribution (Ahuja et al., 2020; Chang et al., 2020; Arjovsky et al., 2019; Lin et al., 2021).

Recent work in invariance learning follows this idea (Lin et al., 2022; Liu et al., 2021; Creager et al., 2021). In general, invariance learning assumes that there are multiple (mostly discrete) environments in the training dataset. Among the environments, the conditional distribution $P(Y|\boldsymbol{X}_s)$ varies but $P(Y|\boldsymbol{X}_v)$ remains the same. From the perspective of causality, the direct causes of Y are $\boldsymbol{X}_v$ and the other factors are $\boldsymbol{X}_s$ (See Section 2 for more discussion). Invariant learning techniques attempt to train a feature extractor Φ that merely learns $\boldsymbol{X}_v$ without the influence of $\boldsymbol{X}_s$. Existing research indicates that Φ can provably extract $\boldsymbol{X}_v$ under proper conditions with a sufficient number of distinct environments. However, it is prevalent for us to lack en-

vironment partitions in practice (Liu et al., 2021; Creager et al., 2021).

Can we learn invariance when there is no environment partition? Several prior literature attempts to address this issue by making assumptions (inductive bias) on the $\boldsymbol{X}_s$ (e.g., Creager et al. (2021) assumes the ERM methods only learn $\boldsymbol{X}_s$; Liu et al. (2021) assumes that discrepancy of spurious features among clusters are larger than that of invariant features). Whereas it is difficult to check these assumptions because they cannot hold generally since $\boldsymbol{X}_s$ might be anything other than the direct causes of Y. The recent study ZIN (Yong et al., 2022) highlights the challenge of identifiability when such assumptions are unavailable. ZIN demonstrates the ***sufficient*** (and almost necessary) conditions to learn invariance with generated environments partitions based on some *auxliary information* $\boldsymbol{X}_z$, which is chosen by some prior knowledge on the causal graph. The primary condition of $\boldsymbol{X}_z$ is $Y \perp \boldsymbol{X}_z | \boldsymbol{X}_v$. That is, in the causal graph, the path between Y and $\boldsymbol{X}_z$ should be d-separated by $\boldsymbol{X}_v$. Though ZIN provides some correct examples to illustrate how to choose $\boldsymbol{X}_z$, it is still non-trivial to verify such condition in practice because it explicitly requires prior knowledge (i.e., $Y \perp \boldsymbol{X}_z | \boldsymbol{X}_v$) and working condition (i.e., given potential $\boldsymbol{X}_z$). That means the user always require prior knowledge on causal graph and explicitly manually determine which feature should be used as $\boldsymbol{X}_z$. This setting can avoid assumptions on spurious features but require human expert involvement which prevents further large-scale usage of the method.

In this work, we propose to seek for such $\boldsymbol{X}_z$ (satisfying the condition $Y \perp \boldsymbol{X}_z | \boldsymbol{X}_v$) in a total data-driven way and utilize $\boldsymbol{X}_z$ to perform environment partitioning. The core idea is to find $\boldsymbol{X}_z$ that is independent of Y (i.e., $\boldsymbol{X}_z \perp Y$). This relaxation only require the working condition (i.e., $\boldsymbol{X}_z$ existance in $\boldsymbol{X}$) to perform domain-agnostic invariance learning. In Section 3.3, we prove that $\boldsymbol{X}_z \perp Y$ can actually leads to $Y \perp \boldsymbol{X}_z | \boldsymbol{X}_v$ under suitable conditions (the Markov and faithfulness property of causal graph). Based on this theoretical observation, we design an efficient algorithm, TIVA (learning invariance *Through Independent Variables Automatically*) [1] that can learn invariance without any environment partition or prior knowledge. We further theoretically show that TIVA can provably learn invariant features when we find a sufficient number of $\boldsymbol{X}_z$ that is independent of Y (we require that $\boldsymbol{X}_z$ should not be totally independent of the whole causal graph, which will be elaborated in Section 3.3). We demonstrate the superiority of TIVA in a series of synthetic and real-world datasets. Notably, we verify TIVA works effectively in both feature selection (we observe a concatenation of $[\boldsymbol{X}_v, \boldsymbol{X}_s, \boldsymbol{X}_z]$) and feature learning tasks (we observe a scrambled transformation of $[\boldsymbol{X}_v, \boldsymbol{X}_s, \boldsymbol{X}_z]$).

[1]The code is released in GitHub repository TIVA.

We summarize our contributions as follows:

- We propose TIVA to learn invariant features without domain partition. TIVA doesn't impose inductive bias on the spurious feature, require prior knowledge of the invariant feature, or prerequisite additional information in environment segmentation.
- We theoretically demonstrate that TIVA can provably identify the invariant features under suitable conditions.
- We empirically show that TIVA achieves superior performance in several synthetic and real-world datasets and demonstrates high efficiency in practice.

The rest of the paper is organized as follows. We first introduce the related literature of invariant learning in Section 2. Then, we state the preliminaries, propose the TIVA, perform theoretical analysis, and describe the specific algorithm in Section 3. After that, synthetic simulation and real-world datasets evaluation with several baseline methods are performed in Section 4. Finally, we conclude the work and discuss the future direction in Section 5. In Appendix, we provide surrogate algorithm of TIVA, detailed theoretical proofs, synthetic simulation details, multiple ablation studies, and several real world examples.

2. Related Work

Invariant learning is a learning technique to separate the invariant and spurious association between features and targets. In the aforementioned object classification example, the patterns of the object are the invariant features $\boldsymbol{X}_v$, which remain stable correlation with Y across different data distributions. In contrast, the background information are spurious features $\boldsymbol{X}_s$ with an unstable correlation. In general, when the IID assumption is violated, models that rely on invariant features (e.g., patterns) will perform robust inference on OOD data due to stable correlation. In contrast, spurious features $\boldsymbol{X}_s$ can induce dramatic performance reduction (Arjovsky et al., 2019). Finding invariant features can be understood from a causal perspective as learning direct cause to ensure robustness under specific heterogeneity and intervention (Peters et al., 2016). Invariant risk minimization is proposed to learn the optimal invariant association across diverse environment segmentation provided in the training data (Arjovsky et al., 2019). To further explore the capability of IRM, several literatures incorporate multi-objective optimization with game theory (Ahuja et al., 2020; Chen et al., 2022) or perform invariant representation learning through adversarial training using deep neural networks (Chang et al., 2020; Xu and Jaakkola, 2021). However, using deep neural networks with IRM training experience some efficiency issues (Lin et al., 2021). This issue is alleviated by reducing over-fitting through

Bayesian framework (Lin et al., 2022) and stabilizing the training process by performing sample reweighting and optimizing under sparsity constraint (Zhou et al., 2022a;b). In addition to invariant learning methods, distributionally robust optimization (DRO) methods have also been developed to improve the OOD generalization (Ben-Tal et al., 2013; Lee and Raginsky, 2018; Gao et al., 2022; Duchi and Namkoong, 2021; Sagawa et al., 2019). Group DRO (Sagawa et al., 2019) shares a similar problem setup to IRM in that the data is segmented into several divergent groups. Robust learning with OOD generalization can be achieved by training on the worst-case loss among all data groups (Sagawa et al., 2019).

Nevertheless, the aforementioned methods require that the data heterogeneity is explicitly represented in a given environment segmentation. In practice, such clear and accurate environment indexes are always unavailable (Creager et al., 2021; Liu et al., 2021), which motivate multiple work to perform invariance learning when the environment partition is not explicitly given by the input data. Sohoni et al. (2020) leverage clustering technique to estimate subgroup labels for the training data and then incorporates these pseudo labels as noisy supervision in a distributionally robust optimization objective. Environment inference for invariant learning (EIIL) (Creager et al., 2021) is proposed to solve this issue in two-stage training by a biased model environment inference and subsequential invariance learning on the environment inference. Heterogeneous risk minimization (HRM) (Liu et al., 2021) tackles this issue by jointly learning latent heterogeneity and invariance with the identification module and invariant predictor respectively. Learning from failure (LfF) (Nam et al., 2020) proposes a failure-based debiasing scheme that assumes the learning difficulty of the spurious information is lower than the invariant feature and spurious information can be captured in the early model learning stage. Therefore, training one biased model and focusing on data with a contradictory prediction result compared to the true label can lead to the second model discovering an invariant correlation (Nam et al., 2020). ZIN (Yong et al., 2022) demonstrates that learning solely on input data without any information about environment partition is theoretically impossible. Additional auxiliary variables provided by metadata of the dataset can be used for efficient environment inference and subsequent invariant learning.

However, in practice, it is still difficult to leverage the aforementioned methods in invariant learning due to the requirement of additional information (Yong et al., 2022) or strong assumption on data distribution and learning process (Creager et al., 2021; Liu et al., 2021; Nam et al., 2020) which may not be practically held. In this paper, we propose TIVA with theoretical guarantee to perform invariant learning, that neither explicitly requires environment partition information, nor prerequisites strong prior knowledge of data distribution. Our algorithm is also not violated the impossibility results analyzed by Yong et al. (2022).

3. Method

3.1. Preliminaries

Throughout this paper, we use upper-cased letters ($\boldsymbol{X} \in \mathbb{R}^d$, $Y \in \mathbb{R}$) to denote random variables, and lower-cased letters to $\boldsymbol{x}$ and y denote deterministic instances. We denote invariant and spurious features as $\boldsymbol{X}_v \in \mathbb{R}^{d_v}$ and $\boldsymbol{X}_s \in \mathbb{R}^{d_s}$,respectively. Suppose the target Y is generated from $\boldsymbol{X}_v$ by a non-degenerate function g_v with an independent random noise: $Y = g_v(\boldsymbol{X}_v, \epsilon_v)$, where $\boldsymbol{X}_v \perp \epsilon_v$.

We also assume there exists a third kind of feature $\boldsymbol{X}_z \in \mathbb{R}^{d_z}$, which is not invariant or spurious. $\boldsymbol{X}_z$ is the feature that not related with the Y prediction task (the auxiliary information in Yong et al. (2022) is an instance of $\boldsymbol{X}_z$). We observe the feature $\boldsymbol{X}$ generated from $\boldsymbol{X}_v$, $\boldsymbol{X}_s$ and $\boldsymbol{X}_z$ with some unknown function q as

$$\boldsymbol{X} = q(\boldsymbol{X}_v, \boldsymbol{X}_s, \boldsymbol{X}_z). \tag{1}$$

The probability of Y under the condition spurious feature $\boldsymbol{X}_s$: $P(Y|\boldsymbol{X}_s)$ can change in different environment (domain) $e \in \mathcal{E}$; while $P(Y|\boldsymbol{X}_v)$ remains invariant. Our goal is to extract $\boldsymbol{X}_v$ from $\boldsymbol{X}$. Following ZIN (Yong et al., 2022), we consider that we normally collect the data from a mixture of environments, and the marginal distribution can be achieved by $P(X, Y) = \sum_{e\in\mathcal{E}} \alpha^e P(\boldsymbol{X}^e, Y^e)$ where $\alpha^e \in [0, 1]$ and $\sum_e \alpha^e = 1$.

Property 1 (Invariance Property)**.** *$P(Y|\boldsymbol{X}_v)$ remains invariant under any intervention on any nodes of the causal graph except for Y itself.*

We aim to learn a function f to predict Y based on $\boldsymbol{X}$. Following Arjovsky et al. (2019), we assume f is composed of a feature extractor Φ and label classifier w, i.e., $f(\boldsymbol{X}) = w(\Phi(\boldsymbol{X}))$. The goal of invariant learning is to learn a feature representation $\Phi(\boldsymbol{X})$ to predict Y that is merely based on $\boldsymbol{X}_v$.

Yong et al. (2022) shows that invariance is unlearnable generally without environmental partition. Specifically, it demonstrates if the environment partitions are not provided, property 1 is insufficient for the identification of $\boldsymbol{X}_v$. In addition, existing work typically makes assumptions (adding inductive bias) on the spurious features (Creager et al., 2021) or requires prior knowledge on the causal graph. In Section 3.2, we explore the ***sufficient*** condition to identify $\boldsymbol{X}_v$ without adding inductive bias or requiring prior knowledge.

3.2. Learning Invariance Through Independent Variables Automatically

Here, we first consider partitioning the dataset into environments by learning a function $\rho(\cdot) : \mathbb{R}^d \to \mathbb{R}^K$

which takes $\boldsymbol{X}$ as the input and output and segment environment in K classes. Here K is a pre-specified number and we denote the k-th entry of $\rho(\cdot)$ as $\rho^{(k)}(\cdot)$. We have $\rho(\boldsymbol{X}) \in [0,1]^K$ and $\sum_k \rho^{(k)}(\boldsymbol{X}) = 1$. Let $\mathcal{R}(w,\Phi) = \frac{1}{n}\sum_{i=1}^n \ell\left(f_w(\Phi(\boldsymbol{x}_i)), y_i\right)$ denote the ERM loss on the total dataset with n total data points. We use $\mathcal{R}^k_\rho(w,\Phi) = \frac{1}{n}\sum_{i=1}^n \rho^{(k)}(\boldsymbol{x}_i)\ell(f_w(\Phi(\boldsymbol{x}_i)), y_i)$ to denote the loss of the k-th inferred environment. Further we assume the function ρ is composed of a environmental feature extractor $u : \mathbb{R}^d \to \mathbb{R}^u$ and environmental index classifier $v : \mathbb{R}^u \to \mathbb{R}^K$, i.e.,

$$\rho_{u,v}(\boldsymbol{X}) = v(u(\boldsymbol{X})). \tag{2}$$

The intuition of considering such structures will be clear in the later part.

IRM (Arjovsky et al., 2019) aims to learn invariant features extractor Φ which elicit an classifier w that is simultaneously optimal in all environments. To achieve this goal, we fit a shared classifier w on the mixture of domains and a set of classifiers $\{w_i\}_{i=1}^K$ (one for each individual inferred environment). Given $\rho_{u,v}$, the invariance penalty is

$$\min_{w,\Phi}\max_{\{w_k\}} \mathcal{L}(\Phi, w, w_1, \cdots, w_K, u, v) := \mathcal{R}(w,\Phi) + \lambda \underbrace{\sum_{k=1}^K \left[\mathcal{R}^k_{\rho_{u,v}}(w,\Phi) - \mathcal{R}^k_{\rho_{u,v}}(w_k,\Phi)\right]}_{\text{invariance penalty}}. \tag{3}$$

The penalty shown in Eqn (3) is widely adopted in IRM literature (Chang et al., 2020; Lin et al., 2022). Following Yong et al. (2022), we adopt the following minimax procedure to automatically learn the environmental partition in an adversarial way:

$$\min_{\omega,\Phi}\max_{u,v,\{\omega_1,\cdots,\omega_K\}} \mathcal{L}(\Phi, \omega, \omega_1, \cdots, \omega_K, u, v). \tag{4}$$

Eqn (4) attempts to find an environmental partition where the spurious features elicit the maximum penalty.

Remark 1. *Eqn (4) is similar to Eqn (6) of Yong et al. (2022). The main difference is that $\rho_{u,v}$ in Eqn (4) takes $\boldsymbol{X}$ as input and ρ in Yong et al. (2022) takes special auxiliary information (carefully chosen based on the prior knowledge of causal graph). Our goal is to relax this requirement on the prior knowledge while still achieving identifiability. Here, we have specific requirement of $u(\boldsymbol{X})$ and it is discussed in the following section.*

In Section 3.3, we first explore the sufficient conditions to identify the invariant feature by Eqn (4) without prior knowledge. Based on the theoretical insights, we propose practical algorithms in Section 3.4.

3.3. Theoretical Analysis

3.3.1. What Information Do We Need?

In this section, we first assume the features $\boldsymbol{X}$ are given, i.e., q in Eqn 1 is an identity mapping. In other words, we have $\boldsymbol{X} = [\boldsymbol{X}_v, \boldsymbol{X}_s, \boldsymbol{X}_z]$. In this case, the environmental feature extractor u of ρ and the label feature extractor Φ of f are both feature masks, i.e., $u \in \{0,1\}^d$ and $\Phi \in \{0,1\}^d$. In Section 3.4, we also extend feature selection to feature learning. It is easy to show that, similar to Yong et al. (2022), if $u(\boldsymbol{X})$ satisfies the following conditions, Eqn. (4) can provably learn invariant features.

Condition 1 (Invariance Preserving Condition)**.** *Given invariant feature $\boldsymbol{X}_v$ and any environmental index classifier $v(\cdot)$, it holds that $H(Y|\boldsymbol{X}_v, v(u(\boldsymbol{X}))) = H(Y|\boldsymbol{X}_v)$.*

Condition 2 (Non-invariance Distinguishing Condition)**.** *For any feature $\boldsymbol{X}_s^k \in \boldsymbol{X}_s$, there exists an environmental index classifier $v(\cdot)$ and a constant $C > 0$ such that $H(Y|\boldsymbol{X}_s^k) - H(Y|\boldsymbol{X}_s^k, v(u(\boldsymbol{X}))) \geq C$.*

We then present the following assumptions, which are identical to Assumption 1-3 of Yong et al. (2022).

Assumption 1. *For a given feature mask Φ and any constant $\epsilon > 0$, there exists $f \in \mathcal{F}$ such that $\mathbb{E}[\ell(f(\Phi(\boldsymbol{X})), Y)] \leq H(Y|\Phi(\boldsymbol{X})) + \epsilon$.*

Assumption 2. *If a feature violates the invariance constraint, adding another feature will not eliminate the penalty, i.e., there exists a constant $\delta > 0$ so that for spurious feature $\boldsymbol{X}_1 \subset \boldsymbol{X}_s$ and any feature $\boldsymbol{X}_2 \subset \boldsymbol{X}$, $H(Y|\boldsymbol{X}_1, \boldsymbol{X}_2) - H(Y|\rho(\boldsymbol{Z}), \boldsymbol{X}_1, \boldsymbol{X}_2) \geq \delta\left(H(Y|\boldsymbol{X}_1) - H(Y|\rho(\boldsymbol{Z}), \boldsymbol{X}_1)\right)$.*

Assumption 3. *Let $\boldsymbol{X}_{-v}$ denote any proper subset of invariant features, i.e., $\boldsymbol{X}_{-v} \subsetneq \boldsymbol{X}_v$, then $H(Y|\boldsymbol{X}_v) \leq H(Y|\boldsymbol{X}_{-v}) - \gamma$ with fixed $\gamma > 0$.*

Now, given the aforementioned conditions and assumptions, we have the following corollary of the Theorem 2 of Yong et al. (2022).

Corollary 1 (Identifiability of Invariant Features)**.** *With Assumptions 1-3 and Conditions 1-2, if $\epsilon < \frac{C\gamma\delta}{4\gamma + 2C\delta H(Y)}$ and $\lambda \in [\frac{H(Y)+1/2\delta C}{\delta C - 4\epsilon} - \frac{1}{2}, \frac{\gamma}{4\epsilon} - \frac{1}{2}]$, then we have $\hat{\mathcal{L}}(\Phi_v) < \hat{\mathcal{L}}(\Phi)$ for all $\Phi \neq \Phi_v$, where $H(Y)$ denotes the entropy of Y. Thus, the solution to Problem 4 identifies invariant features.*

In this section, we restate the conditions and identifiability results of Yong et al. (2022).

Remark 2. *A notable difference is that Yong et al. (2022) requires that the input $\boldsymbol{Z}$ to the environment partition function ρ should satisfy $Y \perp \boldsymbol{Z}|\boldsymbol{X}_v$, which is difficult to check*

if we do not have prior knowledge on the $\boldsymbol{X}_v$. In contrast, in Eqn 2-4 and Condition 1, the environment partition function $\rho_{u,v}$ takes the raw $\boldsymbol{X}$ as input, which does not need any prior knowledge on $\boldsymbol{X}_v$. The key challenge in our framework is how to learn an environmental feature extractor $u(\cdot)$ to satisfy the Condition 1 automatically, which will be discussed in Section 3.3.2.

3.3.2. TIVA: How to Learn Invariance Through Independent Variables Automatically

Condition 1 is equivalent to $Y \perp u(\boldsymbol{X})|\boldsymbol{X}_v$ (the proof is analogous to the proof in Appendix B.6 of Yong et al. (2022) by replacing $\boldsymbol{Z}$ with $u(\boldsymbol{X})$). As is discussed in the section before, $Y \perp u(\boldsymbol{X})|\boldsymbol{X}_v$ is hard to check. In this part, we try to bypass this difficulty by designing $u(\cdot)$. We first provide the following critical lemma:

Lemma 1. *Consider that $(\boldsymbol{X}, Y)$ is generated according to a structural causal model (SCM) with some directed acyclic causal graph $\mathcal{G}$. Assume that both the Markov and faifulness properties hold, i.e., conditional independences wrt. the causal graph (indicated by d-separations) reflect conditional independences wrt. the distributions, and vice versa. Then if $Y \perp u(\boldsymbol{X})$, we have $Y \perp u(\boldsymbol{X}) \mid \boldsymbol{X}_v$, which indicates that Condition 1 holds.*

Proof. Let Y_{nd} be the set of non-descendants of Y, which are the set of nodes that are not reachable from node Y via directed paths. Then by local Markov property, we know $Y \perp X' \mid \boldsymbol{X}_v$ for any $X' \in Y_{nd}$. If $u(\boldsymbol{X}) \perp Y$, then $u(\boldsymbol{X})$ cannot be a descendant of Y, so $X \in Y_{nd}$ and further $Y \perp u(\boldsymbol{X}) \mid \boldsymbol{X}_v$. □

If the data $(\boldsymbol{X}, y)$ are generated following an SCM, then the Markov property will always hold (Spirtes et al., 2000; Pearl, 2009; Peters et al., 2017). The faithfulness assumption is commonly used in constraint-based causal discovery methods that use conditional independences in the data to estimate the underlying causal graph; a recent review can be found in Glymour et al. (2019).

Lemma 1 shows that $Y \perp u(\boldsymbol{X})$ is a sufficient condition for $Y \perp u(\boldsymbol{X})|\boldsymbol{X}_v$. An appealing property here is that $Y \perp u(\boldsymbol{X})$ is easy to check while $Y \perp u(\boldsymbol{X})|\boldsymbol{X}_v$ needs prior knowledge on the causal graph. Motivated by this result, we propose a framework to learn invariance without domain partition in an end-to-end way. We then following formulation by recalling that $Y \perp u(\boldsymbol{X})$ is equivalent to $I(Y, u(\boldsymbol{X})) = 0$, where $I(\cdot,\cdot)$ is the mutual information:

$$\min_{\omega,\Phi} \max_{u,v,\{\omega_1,\cdots,\omega_K\}} \mathcal{L}(\Phi, \omega, \omega_1, \cdots, \omega_K, u, v), \quad s.t. \quad I(u(\boldsymbol{X}), Y) = 0. \tag{5}$$

Theorem 1 (Identifiability of TIVA)**.** *Suppose there exists a subset $\boldsymbol{X}_\perp$ of $\boldsymbol{X}$ that is independent of $\boldsymbol{Y}$. Furthermore, the spurious feature are correlated with $\boldsymbol{X}_\perp$, i.e., for any feature $\boldsymbol{X}_s^k \in \boldsymbol{X}_s$, there exists an environmental index classifier $v(\cdot)$ and a constant $C > 0$ such that $H(Y|\boldsymbol{X}_s^k) - H(Y|\boldsymbol{X}_s^k, v(\boldsymbol{X}_\perp)) \geq C$. With Assumptions 1-3, if $\epsilon < \frac{C\gamma\delta}{4\gamma+2C\delta H(Y)}$ and $\lambda \in [\frac{H(Y)+1/2\delta C}{\delta C-4\epsilon} - \frac{1}{2}, \frac{\gamma}{4\epsilon} - \frac{1}{2}]$, then TIVA in Eqn (5) identifies all invariant features.*

The full theoretical proof is demonstrated in Appendix B.

3.3.3. When will the conditions of TIVA holds

A natural question to ask is when the following condition will hold?

There exists a subset $\boldsymbol{X}_\perp$ of $\boldsymbol{X}$ that is independent of $\boldsymbol{Y}$. Furthermore, the spurious feature are correlated with $\boldsymbol{X}_\perp$.

Figure 1 illustrates several cases when TIVA can successfully identify invariant features without any domain partition or prior knowledge on the causal graph. In the Figure 1(b), we have $X_4 \perp Y$, $X_4 \not\perp Y|X_2$ and $X_4 \not\perp Y|X_3$. Then TIVA will find X_4 and use it to infer environments for invariance learning. An inspiring case, Figure 1(c) shows that even if there is a hidden confounder H and X_4 is not directly linked to spurious features, the conditions can still hold, i.e., $X_4 \perp Y$ and $X_4 \not\perp Y|X_2$.

Connection with Constraint-based Causal Discovery As we previously mentioned, the Markov and faithful assumptions are commonly used in the constraint-based causal discovery methods. Given these two assumptions and a perfect conditional independence test, this class of methods can identify only a set of directed acyclic graphs, the so-called Markov equivalence class, in which the graphs encode the same conditional independences. In particular, all the v-structures can be correctly identified while other edges may not be uniquely determined (Spirtes et al., 2000; Glymour et al., 2019). For example, in the left graph in Figure 1, $X_5 \to X_2 \leftarrow Y$ and $X_4 \to X_3 \leftarrow Y$ can be correctly determined, while the direction between X_1 and Y is not, i.e., both $X_1 \to Y$ and $X_1 \leftarrow Y$ are compatible with conditional independences in the data. Nonetheless, it seems that we can employ these causal discovery approaches to identify more conditional independences than the proposed method, which may further enhance the OOD generalization performance. In our tasks, however, the variables in $\boldsymbol{X}$ may not be explicitly specified; for instance, in the image classification task, the input pixels are unlikely to constitute the semantic causal variables. Even though $\boldsymbol{X}$ represent causal variables, performing conditional independence tests in the non-parametric setting is typically difficult(Shah and Peters, 2020).

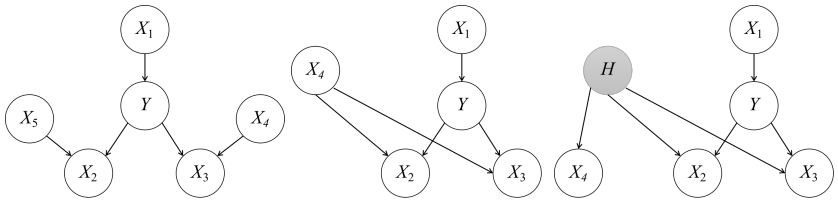


Figure 1. An illustration of cases that satisfy the condition in Theorem 1. In three examples, Y is the target of interest, X_1 is the invariant feature, and X_2 and X_3 are spurious features. The model may use X_4 as X_z. (a) $X_4 \perp Y$, $X_5 \perp Y$, $X_4 \not\perp Y|X_3$ and $X_5 \not\perp Y|X_2$. (b) $X_4 \perp Y$, $X_4 \not\perp Y|X_2$ and $X_4 \not\perp Y|X_3$. (c) H is a hidden confounder. We have $X_4 \perp Y$, $X_4 \not\perp Y|X_2$ and $X_4 \not\perp Y|X_3$.

3.4. Algorithm

In Section 3.2 and 3.3, we present the fundamental concept of TIVA and demonstrate that we can perform invariance learning by Eqn. (5) without providing prior knowledge on data or making additional assumptions. However, directly optimize the Eqn. (5) is difficult. Instead, we propose two-stage surrogate learning algorithm which learns u, v, ω, and Φ by minimizing $\mathcal{R}(\omega, \Phi)$ and $I(Y, u(\boldsymbol{X}))$ in stage one, and minimize the invariance penalty at stage two. The full surrogate algorithm is presented in Appendix A.

The objective is to learn the feature extractor u which satisfies $Y \perp u(\boldsymbol{X})$ and is equivalent to $I(Y, u(\boldsymbol{X})) = 0$, followed by the invariant risk minimization outlined in Section 3.2. In this section, we present two distinct u learning algorithms by taking into account various input feature types which are usually encountered in practice. We conduct ablation studies in Appendix D.2 to evaluate the mutual information during both feature learning and feature selection training. We also investigate the average score of feature selector on $\boldsymbol{X}_v$, $\boldsymbol{X}_s$, and $\boldsymbol{X}_z$.

3.4.1. Feature Selection Algorithm

For normal tabular numerical feature input, we can utilize l_0-based regularization to discriminate high association features $[\boldsymbol{X}_v, \boldsymbol{X}_s]$ and $\boldsymbol{X}_z$. However, l_0 norm cannot be optimized directly by gradient descent methods. Several literatures (Maddison et al., 2016; Jang et al., 2016) have advocated adopting a continuous approximation of discrete random variables, such as the Concrete or Hard-Concrete model. To further reduce the variance in the feature selection, Yamada et al. (2020) developes stochastic gate (stg) by performing Gasussian-based continuous relaxation on Bernoulli variables. We denote the gate $\boldsymbol{U}^+ \in \mathbb{R}^d$ and $\boldsymbol{U}^+ = \max(0, \min(1, \boldsymbol{\mu_u} + \boldsymbol{\epsilon_u}))$ where $\boldsymbol{\epsilon_u}$ is sampled from $\mathcal{N}(0, \boldsymbol{\sigma}^2)$ with fixed $\boldsymbol{\sigma}$. The stg can be directly optimized by model h_ω with parameter ω:

$$\min_{\omega,\mu} \frac{1}{n} \sum_{i=1}^{n} \ell\left(h_\omega(\boldsymbol{x}_i \odot \boldsymbol{U}^+), y_i\right) + \beta ||\boldsymbol{U}^+||_0, \tag{6}$$

where ℓ represents expected ERM loss on data point i, β is the weight parameter, and $\odot$ denotes element-wise production. Yamada et al. (2020) demonstrate that stg learning procedure is equivalent to maximize the mutual information subject to $|\boldsymbol{U}^+|$ equals to a constant. Therefore, to select features that have less mutual information with target $\boldsymbol{Y}$, we simply inverse the stg after the optimization in Eqn. (6) and we can acquire

$$u(\boldsymbol{X}) = \boldsymbol{X} \odot (\boldsymbol{1} - \boldsymbol{U}^+). \tag{7}$$

3.4.2. Feature Learning Algorithm

For images, long sequence input, and other multimodal input features, feature selection may also experience high variance issues (Yamada et al., 2020). We typically perform subsequent tasks on latent embeddings generated by pre-trained large models, especially when implementing image-related tasks or employing large language models. For such application, we develop a feature disentanglement framework to learn features that satisfies $Y \perp u(\boldsymbol{X})$. Similar to Pan et al. (2021), we use one model $\boldsymbol{S} = u_{w_u}(\boldsymbol{X})$ to get irrelevant information $\boldsymbol{S}$ with parameter w_u, one separate model $\boldsymbol{T} = q_{w_q}(\boldsymbol{X}, \boldsymbol{Y})$ to encode the relevant information $\boldsymbol{T}$ with parameter w_q, one decoder $p_{w_p}(\boldsymbol{S}, \boldsymbol{T})$ to reconstruct full input $\boldsymbol{X}$ with parameter w_p, and one discriminator $o_{w_o}(\boldsymbol{S})$ to predict target $\boldsymbol{Y}$ with parameter w_o. These models can be constructed by typical multi-layer perceptron or other complicated network architectures.

We acquire the disentanglement feature by training the aforementioned four models in an adversarial training manner:

$$\begin{aligned}\min_{\omega_u,\omega_p,\omega_q} \max_{\omega_o} \frac{1}{n} \sum_{i=1}^{n} \ell(p_{\omega_p}(u_{w_u}(\boldsymbol{x}_i), q_{w_q}(\boldsymbol{x}_i, y_i)), \boldsymbol{x}_i) \\ - \frac{\lambda}{n} \sum_{i=1}^{n} \ell(o_{w_o}(u_{w_u}(\boldsymbol{x}_i)), y_i),\end{aligned} \tag{8}$$

where ℓ represents the ERM loss on data point i. This learning process can ensure the $\boldsymbol{S}$ has less information with $\boldsymbol{Y}$ but still contain sufficient information with $\boldsymbol{X}$. This satisfies the condition mentioned in Section 3.3.3. After

training, we can acquire irrelevant features through feature encoding $\boldsymbol{S} = u(\boldsymbol{X})$.

4. Experiments

In this section, we perform method evaluation on both synthetic and real-world datasets. This evaluation will also experimentally tests our theoretical analysis discussed in Section 3.3. For baseline methods, we choose ERM to demonstrate the typical OOD performance under IID assumption, IRM (Arjovsky et al., 2019) and group DRO (Sagawa et al., 2019) with given ground-truth environment segmentation to acquire the best OOD performance. We choose HRM (Liu et al., 2021), EIIL (Creager et al., 2021), ZIN (Yong et al., 2022), and LfF (Nam et al., 2020) to compare with our proposed method since these algorithms are also designed to perform invariance learning without providing environment partition. Notice that the ZIN requires further auxiliary information in the dataset (Yong et al., 2022) and LfF is solely designed for classification tasks (Nam et al., 2020). The full hyperparameter settings and model details are demonstrated in Appendix E. TIVA-f denotes the method of feature learning, while TIVA-s represents the method of feature selection.

4.1. Synthetic Dataset

We first test our proposed method in the synthetic dataset by manually setting both invariant and spurious features under a temporal heterogeneity setup. In this dataset, the data will experience a distributional shift with respect to time index $t \in [0, 1]$. We sample the $\boldsymbol{X}_v$, Y, and $\boldsymbol{X}_s$ sequentially to simulate both invariant and spurious correlation. The exhaustive details of sampling process are introduced in Appendix C.

The simulation results are shown in Table 1. Here, we report the average test accuracy and the worst test accuracy among four test environments defined by $p_s(t) = (0.999, 0.8, 0.2, 0.1)$. We mark the highest accuracy among the methods without using environment partitions in bold. We can observe that without any invariant learning capability, ERM method achieves low test accuracy in our synthetic data setting, especially in the worst test accuracy case. This indicates the ERM's limited generalization capability under distributional shift. Our proposed method with feature selection achieves almost identical performance with IRM and ZIN without using any given prior knowledge in environment segmentation in $p_s = (0.999, 0.7)$ and $p_s = (0.999, 0.8)$. On average, our proposed method with feature selection outperforms the baseline method by over around 15%, 19%, and 28% on ERM, EIIL, and HRM with mean accuracy and over 40%, 22%, and 19% with worst accuracy, respectively, which is an significant improvement. We can observe that the performance of the feature learning approach is often inferior to that of the feature selection method. This may be due to the numerical tabular features having a strong and clear association with targets, which might be corrupted in the learning process of the feature disentanglement process.

4.2. Real World Datasets

In this section, we evaluate the performance of our proposed methods on real-world datasets containing distributional shift issues between train and test data. We select three open-source datasets with different types of input features, including numerical tabular features, images, and sequence inputs. We believe these experiments can demonstrate the versatility of our proposed method in plenty of real-world scenarios.

4.2.1. CelebA Dataset

We perform this experiment based on the open-source dataset CelebA (Liu et al., 2018) which contains face images of celebrities. In this task, we classify the *Smiling* of data which is manually operated to spuriously related with the meta information *Gender*. Different from Yong et al. (2022), here the model doesn't require any auxiliary variables or metainformation as input features. Our proposed method performs invariant learning solely on input images. To simulate real practice in image-related tasks, we deploy a pre-trained ResNet (He et al., 2016) model to first acquire hidden features, and perform subsequent invariance learning using the MLP model. We refer the reader to Appendix E for more experiment details.

The CelebA experiment result is shown in Table 2. The best performance is highlighted in bold. We can observe that, although ERM achieves the highest train accuracy and acceptable mean test accuracy, the worst accuracy is relatively low, which indicates a weak OOD generalization capability. The worst test accuracy of EIIL is slightly better than the ERM but still remains a large gap between oracle IRM method and ZIN by using auxiliary information. In this case, our proposed feature learning method outperforms other methods in test mean and test worst accuracy. This may be due to the disentanglement learning process can successively learn X_z from hidden features and feature selection process cannot access such features by directly filtering the hidden features. In comparison, based on the worst test accuracy, LfF and EIL may not capture the invariant feature in this experiment.

4.2.2. House Price Dataset

In this experiment, we perform real-world house price regression task on house price dataset (https://www.kaggle.com/c/house-prices-advanced-regression-techniques). This dataset contains house price recordings from 1900 to 2000 with 17 numerical features. We normalize the house price based on the

Env Info	$p_s(t)$	(0.999, 0.7)				(0.999, 0.8)				(0.999, 0.9)			
	p_v	0.9		0.8		0.9		0.8		0.9		0.8	
	Test Acc	Mean	Worst	Mean	Worst	Mean	Worst	Mean	Worst	Mean	Worst	Mean	Worst
Auxiliary	ZIN	87.50	85.36	77.85	75.39	86.35	82.91	76.79	72.77	83.71	75.89	73.55	64.69
Partition	IRM	87.57	85.47	77.99	75.65	86.57	83.25	77.00	73.39	83.99	76.48	73.84	65.33
No	ERM	75.37	57.31	59.65	25.81	68.72	41.97	55.90	15.07	60.61	23.39	52.85	7.57
	EIIL	38.41	16.80	64.89	49.15	50.77	46.67	68.36	56.35	61.99	53.81	70.10	59.36
	HRM	50.00	49.99	49.98	49.93	50.00	49.98	50.01	49.99	50.00	49.98	49.99	**49.97**
	TIVA-f	81.03	70.84	71.03	58.23	79.10	76.23	73.60	**72.30**	78.03	54.88	**70.26**	34.28
	TIVA-s	**84.42**	**79.60**	**77.49**	**74.56**	**85.40**	**82.79**	**73.63**	71.44	**78.05**	**68.48**	66.83	34.8

Table 1. Test mean and worst accuracy (%) on six temporal heterogeneity synthetic datasets. Env Info denotes the environment information.

Method	Env Info	Train	Test Mean	Test Worst
ZIN	Auxiliary	83.06	76.29	67.27
IRM	Partition	81.30	78.44	75.03
group DRO	Partition	89.32	74.28	58.11
ERM	No	**90.97**	70.76	47.58
LfF	No	59.89	52.97	44.38
EIIL	No	90.01	71.45	50.48
TIVA-f	No	78.80	**72.60**	**60.32**
TIVA-s	No	79.20	72.05	45.67

Table 2. Test accuracy (%) on CelebA task (Accuracy)

same building year to address the natural value ascent over the years. To test model OOD performance, we perform dataset segmentation based on year to get train data when *year* $\in [1900, 1950]$ and test data when *year* $\in (1950, 2000]$. Different from synthetic and CelebA experiments, we cannot directly observe a well-defined environment partition variable. Hence, the performance of IRM here may not be oracle. In reality, most of the real-world scenarios experience an identical situation here, and no clear prior knowledge of environment partitioning is provided. In the house price experiment, we simply divide the train data into five 10-year segments to perform invariant learning.

Method	Env Info	Train	Test Mean	Test Worst
ZIN	Auxiliary	0.2275	0.3339	0.4815
IRM	Partition	0.1327	0.4456	0.6821
group DRO	Partition	0.1213	0.6887	1.0050
ERM	No	**0.1141**	0.4764	0.6703
HRM	No	0.3466	0.4621	0.5721
EIIL	No	0.6841	0.9625	1.3909
TIVA-f	No	0.2389	**0.3050**	**0.4270**
TIVA-s	No	0.2376	0.3276	0.4290

Table 3. The mean squared error of house price task

The experiment result is shown in Table 3. Here we measure the mean squared error (MSE) of the regression and mark the best performance in bold. Our proposed method with feature learning achieves the best performance with 0.305 in test mean MSE and 0.427 in test worst MSE. This performance is even better than IRM and ZIN with providing build year as auxiliary information. We can also observe that the feature learning approach is also superior to the feature selection approach. This indicates that the build year feature itself is insufficient for performing environment inference learning, and more information should be provided. This information may not only be located in individual numerical features that can be selected via feature selection, but it may also be required to disentangle more information.

4.2.3. Landcover Dataset

We perform final evaluation on Landcover dataset (Gislason et al., 2006; Rußwurm et al., 2020; Xie et al., 2020b) to test our proposed method with sequence input. This task requires the model to classify the land cover types by time series input. In this dataset, we can utilize latitude and longitude information as prior knowledge for environment inference since it is irrelevant to the target prediction. For our proposed method training, we don't specially process this information and treat this information as normal feature input. Here, we perform feature learning and selection on the input data that average the time series data along the time axis and concatenate the location data. For train and test dataset separation, we use non-African locations as train set and African locations as test set to validate the OOD performance. For more experiment details, we refer the readers to Appendix E.

Method	Env Info	IID Test	OOD Test
ZIN	Auxiliary	72.18	66.06
ERM (Xie et al., 2020a)	No	**75.92**	58.31
EIIL	No	72.61	64.79
LfF	No	66.24	61.69
TIVA-f	No	68.46	68.26
TIVA-s	No	68.06	**68.31**

Table 4. Test accuracy (%) on Landcover task

The experiment result is shown in Table 4. EIIL performs well on this challenge, and we hypothesize that its initial stage has learned the obvious spurious features. In conjunction with other results, it is still hard to achieve robust results if the spurious feature learning is not guaranteed in stage one. Our proposed technique with feature selection achieves the best OOD performance, outperforming the ZIN method, which specially processes the specified environment index. These results also indicate that even providing prior knowledge information about the environment may

not be sufficient. We can discover more information about $\boldsymbol{X}_z$ for further effective environment inference.

5. Conclusion and Discussion

In this paper, we propose a new algorithm TIVA, that is capable of learning invariance through independent variables automatically without using any predefined environment index, auxiliary information, or prior knowledge of data. Firstly, we demonstrate the general idea of discovering $\boldsymbol{X}_z$ which satisfies the condition $Y \perp \boldsymbol{X}_z|\boldsymbol{X}_v$ can lead to a fully data-driven way of environment inference. Then, we theoretically prove that $\boldsymbol{X}_z \perp Y$ can actually lead to $Y \perp \boldsymbol{X}_z|\boldsymbol{X}_v$ under the mild condition of Markov and faithfulness property of the causal graph. This removes one of the most important requirements in previous work (Yong et al., 2022) and dramatically decreases the difficulty of finding $\boldsymbol{X}_z$ without giving prior knowledge in and causal theory. This makes our algorithm more efficient and adaptive to real-world scenarios and makes fully data-driven invariant learning possible.

Based on this theoretical result, we design the novel invariant learning algorithm by minimizing mutual information and incorporating feature extractor u. To encounter various feature types in practice, we designed two learning approaches for training u. We evaluate our method in both synthetic and real-world datasets with various baseline methods to demonstrate the significance. In future work, we will validate the algorithm's robustness in domain-specific settings.

References

Kartik Ahuja, Karthikeyan Shanmugam, Kush Varshney, and Amit Dhurandhar. Invariant risk minimization games. In *International Conference on Machine Learning*, pages 145–155. PMLR, 2020.

Martin Arjovsky, Léon Bottou, Ishaan Gulrajani, and David Lopez-Paz. Invariant risk minimization. *arXiv preprint arXiv:1907.02893*, 2019.

Kai Arulkumaran, Marc Peter Deisenroth, Miles Brundage, and Anil Anthony Bharath. Deep reinforcement learning: A brief survey. *IEEE Signal Processing Magazine*, 34(6): 26–38, 2017.

Aharon Ben-Tal, Dick Den Hertog, Anja De Waegenaere, Bertrand Melenberg, and Gijs Rennen. Robust solutions of optimization problems affected by uncertain probabilities. *Management Science*, 59(2):341–357, 2013.

Léon Bottou. Stochastic gradient descent tricks. In *Neural networks: Tricks of the trade*, pages 421–436. Springer, 2012.

Shiyu Chang, Yang Zhang, Mo Yu, and Tommi Jaakkola. Invariant rationalization. In *International Conference on Machine Learning*, pages 1448–1458. PMLR, 2020.

Yongqiang Chen, Kaiwen Zhou, Yatao Bian, Binghui Xie, Kaili Ma, Yonggang Zhang, Han Yang, Bo Han, and James Cheng. Pareto invariant risk minimization. *arXiv preprint arXiv:2206.07766*, 2022.

KR1442 Chowdhary. Natural language processing. *Fundamentals of artificial intelligence*, pages 603–649, 2020.

Elliot Creager, Jörn-Henrik Jacobsen, and Richard Zemel. Environment inference for invariant learning. In *International Conference on Machine Learning*, pages 2189–2200. PMLR, 2021.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805*, 2018.

John C Duchi and Hongseok Namkoong. Learning models with uniform performance via distributionally robust optimization. *The Annals of Statistics*, 49(3):1378–1406, 2021.

Rui Gao, Xi Chen, and Anton J Kleywegt. Wasserstein distributionally robust optimization and variation regularization. *Operations Research*, 2022.

Pall Oskar Gislason, Jon Atli Benediktsson, and Johannes R Sveinsson. Random forests for land cover classification. *Pattern recognition letters*, 27(4):294–300, 2006.

Clark Glymour, Kun Zhang, and Peter Spirtes. Review of causal discovery methods based on graphical models. *Frontiers in Genetics*, 10, 2019.

Jiuxiang Gu, Zhenhua Wang, Jason Kuen, Lianyang Ma, Amir Shahroudy, Bing Shuai, Ting Liu, Xingxing Wang, Gang Wang, Jianfei Cai, et al. Recent advances in convolutional neural networks. *Pattern recognition*, 77:354–377, 2018.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Dan Hendrycks and Kevin Gimpel. A baseline for detecting misclassified and out-of-distribution examples in neural networks. *arXiv preprint arXiv:1610.02136*, 2016.

Biwei Huang, Kun Zhang, Jiji Zhang, Joseph D Ramsey, Ruben Sanchez-Romero, Clark Glymour, and Bernhard Schölkopf. Causal discovery from heterogeneous/nonstationary data. *J. Mach. Learn. Res.*, 21(89): 1–53, 2020.

Eric Jang, Shixiang Gu, and Ben Poole. Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*, 2016.

Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Yann LeCun, Yoshua Bengio, and Geoffrey Hinton. Deep learning. *nature*, 521(7553):436–444, 2015.

Jaeho Lee and Maxim Raginsky. Minimax statistical learning with wasserstein distances. *Advances in Neural Information Processing Systems*, 31, 2018.

Yuanzhi Li and Yang Yuan. Convergence analysis of two-layer neural networks with relu activation. *Advances in neural information processing systems*, 30, 2017.

Shiyu Liang, Yixuan Li, and Rayadurgam Srikant. Enhancing the reliability of out-of-distribution image detection in neural networks. *arXiv preprint arXiv:1706.02690*, 2017.

Yong Lin, Qing Lian, and Tong Zhang. An empirical study of invariant risk minimization on deep models. In *ICML 2021 Workshop on Uncertainty and Robustness in Deep Learning*, volume 1, page 7, 2021.

Yong Lin, Hanze Dong, Hao Wang, and Tong Zhang. Bayesian invariant risk minimization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 16021–16030, 2022.

Jiashuo Liu, Zheyuan Hu, Peng Cui, Bo Li, and Zheyan Shen. Heterogeneous risk minimization. In *International Conference on Machine Learning*, pages 6804–6814. PMLR, 2021.

Ziwei Liu, Ping Luo, Xiaogang Wang, and Xiaoou Tang. Large-scale celebfaces attributes (celeba) dataset. *Retrieved August*, 15(2018):11, 2018.

Scott M Lundberg and Su-In Lee. A unified approach to interpreting model predictions. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4765–4774. Curran Associates, Inc., 2017.

Chris J Maddison, Andriy Mnih, and Yee Whye Teh. The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*, 2016.

Junhyun Nam, Hyuntak Cha, Sungsoo Ahn, Jaeho Lee, and Jinwoo Shin. Learning from failure: De-biasing classifier from biased classifier. *Advances in Neural Information Processing Systems*, 33:20673–20684, 2020.

Ziqi Pan, Li Niu, Jianfu Zhang, and Liqing Zhang. Disentangled information bottleneck. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 9285–9293, 2021.

Judea Pearl. *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan kaufmann, 1988.

Judea Pearl. *Causality*. Cambridge University Press, 2009.

J. Peters, D. Janzing, and B. Schölkopf. *Elements of Causal Inference - Foundations and Learning Algorithms*. The MIT Press, Cambridge, MA, 2017.

Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference by using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 78 (5):947–1012, 2016.

Marc Rußwurm, Sherrie Wang, Marco Korner, and David Lobell. Meta-learning for few-shot land cover classification. In *Proceedings of the ieee/cvf conference on computer vision and pattern recognition workshops*, pages 200–201, 2020.

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

Rajen D Shah and Jonas Peters. The hardness of conditional independence testing and the generalised covariance measure. *The Annals of Statistics*, 48(3):1514–1538, 2020.

Guy Shani and Asela Gunawardana. Evaluating recommendation systems. In *Recommender systems handbook*, pages 257–297. Springer, 2011.

Nimit Sohoni, Jared Dunnmon, Geoffrey Angus, Albert Gu, and Christopher Ré. No subclass left behind: Fine-grained robustness in coarse-grained classification problems. *Advances in Neural Information Processing Systems*, 33:19339–19352, 2020.

Peter Spirtes, Clark N Glymour, and Richard Scheines. *Causation, Prediction, and Search*. MIT Press, second edition, 2000.

Raman Uppal and Tan Wang. Model misspecification and underdiversification. *The Journal of Finance*, 58(6):2465–2486, 2003.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Mei Wang and Weihong Deng. Deep visual domain adaptation: A survey. *Neurocomputing*, 312:135–153, 2018.

Feng Xiao, Y Honma, and T Kono. A simple algebraic interface capturing scheme using hyperbolic tangent function. *International journal for numerical methods in fluids*, 48 (9):1023–1040, 2005.

Chuanlong Xie, Fei Chen, Yue Liu, and Zhenguo Li. Risk variance penalization: From distributional robustness to causality. *arXiv preprint arXiv:2006.07544*, 1, 2020a.

Sang Michael Xie, Ananya Kumar, Robbie Jones, Fereshte Khani, Tengyu Ma, and Percy Liang. In-n-out: Pre-training and self-training using auxiliary information for out-of-distribution robustness. *arXiv preprint arXiv:2012.04550*, 2020b.

Yilun Xu and Tommi Jaakkola. Learning representations that support robust transfer of predictors. *arXiv preprint arXiv:2110.09940*, 2021.

Yutaro Yamada, Ofir Lindenbaum, Sahand Negahban, and Yuval Kluger. Feature selection using stochastic gates. In *International Conference on Machine Learning*, pages 10648–10659. PMLR, 2020.

LIN Yong, Shengyu Zhu, Lu Tan, and Peng Cui. Zin: When and how to learn invariance without environment partition? In *Advances in Neural Information Processing Systems*, 2022.

Xiao Zhou, Yong Lin, Renjie Pi, Weizhong Zhang, Renzhe Xu, Peng Cui, and Tong Zhang. Model agnostic sample reweighting for out-of-distribution learning. In *International Conference on Machine Learning*, pages 27203–27221. PMLR, 2022a.

Xiao Zhou, Yong Lin, Weizhong Zhang, and Tong Zhang. Sparse invariant risk minimization. In *International Conference on Machine Learning*, pages 27222–27244. PMLR, 2022b.

A. Surrogate Learning Algorithm

The objective function shown in Eqn. (5) is hard to optimize due to the minimax formulation. Here, we can perform the first-order approximation on the invariance penalty term and Lagrangian transformation on mutual information constraint. Hence, in practice, we can optimize the model with this surrogate minimax objective function:

$$\min_{\omega,\Phi}\max_{u,v}\mathcal{R}(\omega,\Phi)+\lambda\left[\sum_{k=1}^{K}||\nabla_{\omega}\mathcal{R}_{u,v}(\omega,\Phi)||^2-I(u(\boldsymbol{X}),Y)\right], \tag{9}$$

where $I(u(\boldsymbol{X}),Y)$ denotes the mutual information loss discussed in Section 3.4.2 and 3.4.1. This mutual information can be minimized by Eqn. (6) or Eqn. (8) depends on the feature types. The connection between Eqn (5) and Eqn (9) had been discussed in Appendix B. Overall, in implementation, we can follow a two-stage training manner to optimize the model. The complete algorithm is described in Algorithm 1.

Algorithm 1 TIVA

Input: feature extractor Φ, label classifier ω, environmental feature extractor u, environmental index classifier v. Input data $\boldsymbol{X}$ and target $\boldsymbol{Y}$.
1. Stage-one: Annealing Iteration
for each annealing iteration on minibatch **do**
  if $\boldsymbol{X}$ is tabular numerical concatenation feature **then**
    Train u by minimizing the Eqn. (6) and acquiring new gate by Eqn. (7).
  else
    Train u, by minimizing the Eqn. (8).
  end if
  Train Φ, ω, and v, by minimize ERM loss $\mathcal{R}(\omega,\Phi)$ of ω and Φ, and maximize the $\sum_{k=1}^{K}||\nabla_{\omega}\mathcal{R}_{u,v}(\omega,\Phi)||^2$ over u, v on K partitions.
end for
2. Stage-two: Invariance Learning Iteration
for each training iteration on minibatch **do**
  Train Φ and ω by fixing the u and v, and optimizing the Eqn. (9) over the Φ and ω.
end for

B. Proofs

In this section, we provide the full proof of Theorem 1.

Proof. Given a feature mask $\Phi\in\{0,1\}^d$, we can solve the problem in Eqn (5) and obtain a loss $\hat{\mathcal{L}}(\Phi)$ as

$$\begin{aligned}\hat{\mathcal{L}}(\Phi)=\min_{\omega}\max_{u,v,\{\omega_1,\cdots,\omega_K\}}&\ \mathcal{L}(\Phi,\omega,\omega_1,\cdots,\omega_K,u,v),\\ s.t.\quad &I(u(\boldsymbol{X}),Y)=0.\end{aligned} \tag{10}$$

Let Φ_v denote a feature mask that merely selects the invariant feature $\boldsymbol{X}_v$. Our target is to show that

$$\hat{\mathcal{L}}(\Phi_v)<\mathcal{L}(\Phi),\forall\Phi\neq\Phi_v, \tag{11}$$

which is equivalent to that solving Eqn (5) can uniquely identify $\boldsymbol{X}_v$. Our proof proceeds in two steps. First, we show that any feature mask that selects at least one spurious feature would induce a penalty. With sufficiently large λ, the penalty will dominate the expected risk and then exceed $\hat{\mathcal{L}}(\Phi_v)$. Second, we show that any proper subset of the invariant features induces a loss larger than $\hat{\mathcal{L}}(\Phi_v)$.

Step 1 Suppose that the feature mask contains at least one spurious feature. Denote the selected features as $\boldsymbol{X}_{+s}$ and the corresponding feature mask as Φ_{+s}. We aim to show that

$$\hat{\mathcal{L}}_{(\Phi_{+s})}>\hat{\mathcal{L}}(\Phi_v).$$

Since $I(Y, u(\boldsymbol{X})) = 0$, so $Y \perp u(\boldsymbol{X})$. Then by Lemma 1, we have $Y \perp u(\boldsymbol{X})|\boldsymbol{X}_v$, which is equivalent to $H(Y|\boldsymbol{X}_v) - H(Y|\boldsymbol{X}_v, v(u(\boldsymbol{X}))) = 0$ for any v.

By Assumption 1 with a given $\epsilon > 0$, we have

$$\begin{aligned}\hat{\mathcal{L}}(\Phi_v) &\leq (1+2\lambda)\epsilon + H(Y|\boldsymbol{X}_v) \\ &\quad + \lambda\left(H(Y|\boldsymbol{X}_v) - H(Y|\boldsymbol{X}_v, v(u(\boldsymbol{X})))\right) \\ &= (1+2\lambda)\epsilon + H(Y|\boldsymbol{X}_v) \\ &\leq (1+2\lambda)\epsilon + H(Y).\end{aligned}$$

Let $u_\perp$ denote the environment feature mask that merely select $\boldsymbol{X}_\perp$, i.e., $u_\perp(\boldsymbol{X}) = \boldsymbol{X}_\perp$. Let us consider the loss with fixed Φ and u as following:

$$\begin{aligned}\hat{\mathcal{L}}(\Phi, u) = \min_{\omega} \max_{v, \{\omega_1, \cdots, \omega_K\}} &\mathcal{L}(\Phi, \omega, \omega_1, \cdots, \omega_K, u, v), \\ s.t. \quad &I(u(\boldsymbol{X}), Y) = 0.\end{aligned}$$

We first have $\hat{\mathcal{L}}(\Phi_{+s}) \geq \hat{\mathcal{L}}(\Phi_{+s}, u_\perp)$ because $\hat{\mathcal{L}}(\Phi_{+s})$ takes maximum over all possible u. So we have

$$\begin{aligned}\mathcal{L}(\Phi_{+s}) &\geq \hat{\mathcal{L}}(\Phi_{+s}, u_\perp) \\ &\geq -(1+2\lambda)\epsilon + H(Y|\boldsymbol{X}_{+s}) + \\ &\quad \lambda\left(H(Y|\boldsymbol{X}_{+s}) - H(Y|\boldsymbol{X}_{+s}, v(\boldsymbol{X}_\perp))\right) \\ &\geq -(1+2\lambda)\epsilon + \lambda(H(Y|\boldsymbol{X}_{+s}) \\ &\quad - H(Y|\boldsymbol{X}_{+s}, v(\boldsymbol{X}_\perp)) \\ &\geq -(1+2\lambda)\epsilon + \lambda\delta C,\end{aligned}$$

where the last inequality is due to Assumption 2 and $H(Y|\boldsymbol{X}_s) - H(Y|\boldsymbol{X}_s, v(\boldsymbol{X}_\perp)) \geq C$. Thus, if we choose $\epsilon < \delta C/4$ and $\lambda > \frac{H(Y)+2\epsilon}{\delta C - 4\epsilon}$, we can get

$$\mathcal{L}(\Phi_v) < \mathcal{L}(\Phi_{+s}).$$

Step 2 Let Φ_{-v} denote the feature mask corresponding to $\boldsymbol{X}_{-v}$.

In Step 1, we have shown that

$$\hat{\mathcal{L}}(\Phi_v) \leq (1+2\lambda)\epsilon + H(Y|\boldsymbol{X}_v). \tag{12}$$

Similar to Eqn. (12), we have

$$\hat{\mathcal{L}}(\Phi_{-v}) \geq -(1+2\lambda)\epsilon + H(Y|\boldsymbol{X}_{-v}). \tag{13}$$

Then according to Assumption 3, we have

$$\begin{aligned}\hat{\mathcal{L}}(\Phi_{-v}) - \hat{\mathcal{L}}(\Phi_v) &\geq -2(1+2\lambda)\epsilon + H(y|\boldsymbol{X}_{-v}) \\ &\quad - H(y|\boldsymbol{X}_v) \\ &\geq -2(1+2\lambda)\epsilon + \gamma.\end{aligned} \tag{14}$$

Thus, if $\epsilon < \frac{\gamma}{2(1+2\lambda)}$, we have

$$\hat{\mathcal{L}}(\Phi_{-v}) > \hat{\mathcal{L}}(\Phi_v). \tag{15}$$

In conclusion, with $\lambda \in [\frac{H(Y)+1/2\delta C}{\delta^d C - 4\epsilon} - \frac{1}{2}, \frac{\gamma}{4\epsilon} - \frac{1}{2}]$, we can get

$$\hat{\mathcal{L}}(\Phi_v) < \hat{\mathcal{L}}(\Phi), \quad \forall \Phi \neq \Phi_v.$$

Notably, there exists a feasible λ if $\epsilon < \frac{C\gamma\delta}{4\gamma + 2C\delta H(Y)}$. The proof is complete by noticing that ϵ can be chosen arbitrarily according to Assumption 1. □

C. Synthetic Simulation

We first sample invariant feature $X_v(t) \in \mathbb{R}$ from two normal distribution with identical probability:

$$X_v(t) \sim \begin{cases} \mathcal{N}(1,1), w.p. & 0.5, \\ \mathcal{N}(-1,1), w.p. & 0.5. \end{cases} \tag{16}$$

Then, we can sample the target by a consistent probability to induce an invariant correlation between target $Y(t) \in \mathbb{R}$ and $X_v(t)$:

$$Y(t) \sim \begin{cases} \text{sign}(x_v(t)), w.p. & p_v, \\ -\text{sign}(x_v(t)), w.p. & 1-p_v. \end{cases} \tag{17}$$

To model the distributional shift w.r.t. time, we set different $p_s(t)$ in separate isometric time intervals to acquire a spurious correlation between $Y(t)$ and $X_s(t)$:

$$X_s(t) \sim \begin{cases} \mathcal{N}(Y(t),1), w.p. & p_s(t), \\ \mathcal{N}(-Y(t),1), w.p. & 1-p_s(t). \end{cases} \tag{18}$$

The $p_s(t)$ should be set differently between the train dataset and test dataset but p_v should remain the same. For training, we utilize two $p_s(t)$ on time interval set: $\{[0, 0.5), [0.5, 1]\}$. For testing, we designed four $p_s(t)$ on time interval set: $\{[0, 0.25), [0.25, 0.5), \{0.5, 0.75), [0.75, 1]\}$. That means we test our method by training the model on two heterogeneous environment segments and evaluating on four segmentation. We denote the various p_s in tuple. For example, the first simulation is performed on $(0.999, 0.7)$ which stands for:

$$p_s(t) = \begin{cases} 0.999, & t \in [0, 0.5), \\ 0.7, & t \in [0.5, 1]. \end{cases} \tag{19}$$

For both train and test dataset, we directly set $X_z = t$. This feature is also common in practice, for example, we may use data index or user id as input features in recommendation systems (Shani and Gunawardana, 2011). The simulation setting is identical to the synthetic setting reported in Yong et al. (2022), so the result is comparable. We refer readers to Appendix E For full implementation details of this simulation.

D. Ablation Study

D.1. Satisfaction of Condition 2

To evaluate the satisfaction of condition 2, we conducted an ablation study in the synthetic dataset introduced in the Section 4. We approximate the $H(Y|\boldsymbol{X}_s)$ and $H(Y|\boldsymbol{X}_s, v(u(\boldsymbol{X})))$ with cross-entropy loss using neural networks, and calculated their difference to obtain the value $H(Y|\boldsymbol{X}_s) - H(Y|\boldsymbol{X}_s, v(u(\boldsymbol{X})))$. We approximate this value on the test set, and we believe that this approach provides a rigorous and reliable evaluation of satisfaction of condition 2. The ablation study result is shown in the Table 5.

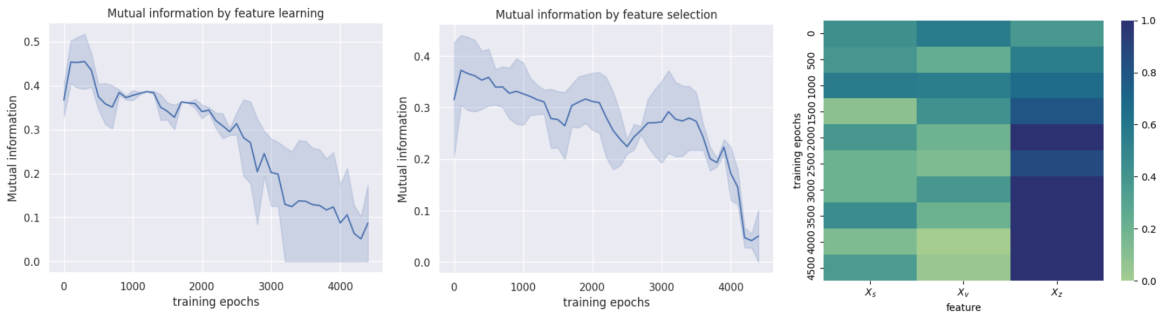
$p_s(t)$	(0.999, 0.7)		(0.999, 0.8)		(0.999, 0.9)	
p_v	0.9	0.8	0.9	0.8	0.9	0.8
Condition 2	0.7894	0.5662	0.8393	0.8346	1.3110	1.2658

Table 5. The approximate value of condition 2

Based on our ablation study, we can observe that condition 2 is indeed satisfied.

D.2. Mutual Information

In this section, we perform ablation study on mutual information learning process to test whether the feature selection and feature learning algorithm mentioned in Section 3.4.1 and Section 3.4.2 can learn $\boldsymbol{X}_z \perp Y$ or not. We perform the ablation study in the synthetic simulation setting $p_s(t) = (0.999, 0.7)$ with $p_v = 0.9$ and measure the mutual information $I(u(\boldsymbol{X}), Y)$. The mutual information is approximated by $I(u(\boldsymbol{X}), Y) = H(Y) - H(Y|u(\boldsymbol{X}))$, where $H(Y)$ denotes the entropy of labels and $H(Y|u(\boldsymbol{X}))$ represents the conditional entropy of the labels given $u(\boldsymbol{X})$ (Pan et al., 2021). $H(Y)$



(a) (b) (c)

Figure 2. Ablation study results of both learning methods.

is fixed when the dataset is determined. We use neural networks to approximate the $H(Y|u(\boldsymbol{X}))$ by calculating the final cross-entropy loss after training. We follow all the settings mentioned in Section 4 and Appendix C. We report the mutual information $I(u(\boldsymbol{X}), Y)$ of both feature selection and feature learning method by each 100 epoch of training in Figure 2.

As shown in Figure 2(a) and Figure 2(b), both feature learning and feature selection method can gradually reduce the mutual information between $\boldsymbol{X}_z$ candidates and Y. In this experiment, feature selection method can achieve lower and more stable mutual information than feature learning methods, which also support the experiment results in Table 1 that feature selection can generally achieve higher invariance learning capability in synthetic simulation. Figure 2(c) demonstrate the heat-map of feature selection gate average score on $u(\boldsymbol{X}_v)$, $u(\boldsymbol{X}_s)$, and $u(\boldsymbol{X}_z)$, respectively. We can observe that, during the training, feature selector u mainly select $\boldsymbol{X}_z$ for environment inference.

D.3. Influence of K

We perform an ablation analysis on the synthetic dataset to explore the impact of the K value on the efficacy of our proposed algorithm. The result is shown in the Table 6

$p_s(t)$	(0.999, 0.7)		(0.999, 0.8)		(0.999, 0.9)	
p_v	0.9	0.8	0.9	0.8	0.9	0.8
$K = 2$	84.42%/79.60%	77.49% /74.56%	85.40% /82.79%	73.63% /71.44%	78.05% / 68.48%	70.26% / 34.8%
$K = 4$	84.86% /78.40%	77.76% /76.56%	83.40% /79.96%	72.40% /69.48%	75.08% /66.42%	69.92%/ 39.12%
$K = 6$	84.32% /78.94%	77.23% /75.42%	83.96% /80.12%	72.38% /70.54%	75.28% /67.58%	69.13% /40.24%
ERM	75.37% /57.31%	59.65% /25.81%	68.72% / 41.97%	55.90% /15.07%	60.61% /23.39%	52.85% /7.57%

Table 6. The ablation study results of value K on synthetic dataset. Here we report the mean and worst accuracy for each test.

The ablation results show that our algorithm is relatively resilient to the selection of the K value, which is in line with the observations made in ZIN (Yong et al., 2022).

D.4. Illustrate the Learned Environment Partition

To better illustrate the learned the environment partition, we perform ablation study to observe the subgroup of environment partition learning with some of the partition labels used in the ZIN (Yong et al., 2022) on CelebA dataset. In the case of the CelebA test, we are predicting the $Smile$ variable and constructing datasets that are spuriously correlated with $Gender$. To better illustrate the different environments, we perform another ablation study to observe the $Smile$ and $Gender$ subgroup distributions in the learned environments. The ablation result shown in the Table 7

subgroup	$P(Group\|Env)$ in inferred environment 1	$P(Group\|Env)$ in inferred environment 2
$Smile = Gender$	64.62%	36.62%
$Smile \neq Gender$	35.38%	63.38%

Table 7. The ablation study results on the involvement of random noise.

The ablation study results showed that the group distributions of $Smile = Gender$ and $Smile \neq Gender$ are distributed differently in the learned environments. Recall that smile is Y and gender is spurious feature $\boldsymbol{X}_s$. This result shows that $P(Y|\boldsymbol{X}_s)$ differs in the two inferred environments, making it possible for the invariance penalty to distinguish the spurious feature.

D.5. Random Noise as $\boldsymbol{X}_z$

Since random noise is independent of Y, can random noise be used as $\boldsymbol{X}_z$ for TIVA learning? In order to evaluate the influence of random noise in the features, we conduct an ablation study in our synthetic dataset introduced in Section 4 to evaluate its impact on our methodology. Specifically, we add random noise to the dataset and evaluated its utilization by observing the feature selection gate value (i.e., selection by u which is similar to Appendix D.2) and Shapley value (Lundberg and Lee, 2017) to measure the feature contribution of each feature in v. To further evaluate the effectiveness of utilizing random noise as auxiliary information, we conducted an additional ablation study by replacing all t values with random noise, and tested the final experiment value. The ablation study results are summarized in the Table 8

$p_s(t)$	(0.999, 0.7)		(0.999, 0.8)		(0.999, 0.9)	
p_v	0.9	0.8	0.9	0.8	0.9	0.8
gate value	1					
Shapley value of random noise	$9.6e^{-5}$	$5.4e^{-5}$	$7.3e^{-5}$	$1.3e^{-5}$	$2.3e^{-5}$	$3.8e^{-5}$
Shapley value of $\boldsymbol{X}_z$	0.4947	0.4996	0.4967	0.4972	0.4939	0.4991
mean accuracy of solely random noise	74.04%	59.28%	66.98%	55.40%	59.22%	52.94%
worst accuracy of solely random noise	55.04%	26.64%	40.08%	16.72%	22.80%	9.44%
worst accuracy of ERM	57.31%	25.81%	41.97%	15.07%	23.39%	7.57%

Table 8. The ablation study results on the involvement of random noise.

Based on the ablation study results, we can observe that although u selects random noise as a candidate for $\boldsymbol{X}_z$ as the end of the training epoch, the model v does not choose random noise as auxiliary information, and the Shapley value on random noise is very small. Furthermore, our ablation study demonstrated that the model is unable to utilize random noise as auxiliary information because it does not contain any information about $\boldsymbol{X}_s$. It is important to note that if we cannot find any useful information in $\boldsymbol{X}$, the model reduces to ERM.

D.6. In-distribution and Out-of-distribution Trade-off

In the general cases of implementing invariance learning, we are exploiting the trade-off between in-distribution and out-of-distribution performance. That means, the performance drop of in-distribution performance is somehow inevitable. Invariant learning seeks to identify the stable and consistent features $\boldsymbol{X}_v$ that directly influence the target variable Y, while avoiding the use of spurious features $\boldsymbol{X}_s$ that may be inconsistent or unstable across different environments. In practice, utilizing both $\boldsymbol{X}_s$ and $\boldsymbol{X}_v$ can lead to better in-distribution performance because the model has access to more information. However, when the model is tested on out-of-distribution data, the relation between $\boldsymbol{X}_s$ and Y may experience large distributional shift, causing the model's performance to drop dramatically. Thus, a model with high in-distribution accuracy that relies heavily on $\boldsymbol{X}_s$ may be dangerous and lead to poor generalization. This is common especially when Y cannot be deterministically predicted from $\boldsymbol{X}_v$ (e.g., cannot achieve 100% accuracy by using the invariant feature) because there is always randomness in data generation mechanisms and model misspecification. Specifically:

- Randomness in data generation: randomness in data generation mechanisms can result in a scenario where Y cannot be fully predicted by $\boldsymbol{X}_v$. Specifically, it is a common practice to consider the structural causal equations (SEM) (Peters et al., 2017) as $Y = g(\boldsymbol{X}_v) + \epsilon$ where ϵ is random noise (Arjovsky et al., 2019). Assuming $\epsilon = 0$, Y can be deterministically predicted from $\boldsymbol{X}_v$. However, assuming $\epsilon = 0$ is known as the degenerated setting and is not commonly observed in theoretical or practical literature. It is also believed that there is always randomness in data (Pearl, 1988).
- Model misspecification: even in cases where there is no randomness in data generation, Y may still not be perfectly predicted by $\boldsymbol{X}_v$ due to model misspecification (Uppal and Wang, 2003). By "model misspecification", the true function g may not lie in our function class. We can also interpret it as the invariant feature being unable to be perfectly extracted.

In both scenarios shown above, including informative spurious features $\boldsymbol{X}_s$ can improve the model's ability to predict the label accurately. Specifically, we have $H(Y|\boldsymbol{X}_v) > 0$, indicating that Y cannot be fully predicted based on $\boldsymbol{X}_v$. Consequently, we can obtain lower prediction errors by using spurious features in in-distribution tests, resulting in $H(Y|\boldsymbol{X}_v, \boldsymbol{X}_s) < H(Y|\boldsymbol{X}_v)$ (Yong et al., 2022). Therefore, in most real datasets where we cannot perfectly predict Y based on merely invariant features, the in-distribution performance may be slightly inferior when we remove spurious features.

However, there may exist some scenarios that removing the influence of $\boldsymbol{X}_s$ would not harm the in-distribution performance and can further improve the out-of-distribution performance. This situation may occur when Y can be fully predicted by $\boldsymbol{X}_v$.

To better illustrate the aforementioned effect, we perform another ablation study on synthetic dataset. We set $p_v = 0.999$ and $p_s = (0.999, 0.9)$, and the test distribution is $(0.999, 0.9, 0.2, 0.01)$. Under this setting, the test distribution differs slightly from the synthetic experiment in the Section 4, and the target Y was nearly deterministically generated from $\boldsymbol{X}_v$. We test the randomness by setting the $p_v = 0.9$ to simulate the randomness in generating Y from $\boldsymbol{X}_v$. while other parameters remained the same. The ablation results are presented in the Table 9

Methods	Mean	0.01	0.2	0.9	0.999
ERM with $p_v = 0.999$	96.44%	93.52%	94.56%	98.40%	99.28%
TIVA-s with $p_v = 0.999$	97.50%	95.28%	96.72%	98.72%	99.28%
ERM with $p_v = 0.9$	61.08%	22.72%	34.64%	90.24%	96.72%
TIVA-s with $p_v = 0.9$	80.70%	69.68%	74.32%	88.40%	90.40%

Table 9. Test accuracy on each out-of-distribution envrionment partition.

The results show that if Y is almost determined by $\boldsymbol{X}_v$, we can actually indeed improve the OOD performance without harming the in-distribution performance. However, when there is randomness in generating Y, as is commonly observed in practice, we may experience a "sacrifice" in in-distribution performance to improve out-of-distribution performance.

E. Hyperparameter Setting and Experiment Details

In this section, we introduce the experiment details of Section 4. We demonstrate the hyperparameter in Table 10. We perform all the experiments on Nvidia V100 GPU. The celebA experiment takes roughly two GPU hours and other tasks can be completed in 10 minutes.

The part of the hyperparameter used by our proposed method in Section 4 is shown in Table 10. Here we represent the neural network architecture as a list, for example, the v we used across all the experiments is 2 layer MLP with 32 neurons, which is denoted as $[32, 32]$. Here, we use the ReLU activation function across all networks (Li and Yuan, 2017). For encoder u_{ω_u} and q_{ω_q}, the output layer is activated by tanh function (Xiao et al., 2005).

These hyperparameters are finetuned by the grid search method in 5 trials.

Experiment with Method	learning rate	u_{w_u}	q_{w_q}	p_{w_p}	o_{w_o}	β	v	K	o_{w_o} learning rate
synthetic TIVA-f	0.001	$[16, 8]$	$[16, 4]$	$[16, 16]$	$[16, 16]$	-	$[32, 32]$	2	0.01
synthetic TIVA-s	0.01	-	-	-	-	0.05			-
celebA TIVA-f	0.001	$[256, 256]$	$[128, 128]$	$[256, 32]$	$[256, 256]$	-			0.01
celebA TIVA-s	0.001	-	-	-	-	0.01			-
house price TIVA-f	0.001	$[16, 8]$	$[16, 4]$	$[16, 16]$	$[16, 16]$	-			0.02
house price TIVA-s	0.001	-	-	-	-	0.03			-
landcover TIVA-f	0.1	$[16, 8]$	$[16, 4]$	$[16, 16]$	$[16, 16]$	-		6	0.5
landcover TIVA-s	0.1	-	-	-	-	0.03			-

Table 10. Hyperparameter for experiments

In Synthetic simulation, here we utilize one linear layer $[16]$ as Φ with 1024 batch size. We train the model in 5000 epochs with 4500 epoch environment annealing. In celebA tasks, we use large vision model ResNet-18 (He et al., 2016) as the fixed backbone to extract latent image features. We use 128 batch size in 50 total epochs of training with 45 epochs of annealing. For house price task, we implement two layers MLP $[16, 16]$ as Φ and perform 5500 epochs of training with 5000 epochs of annealing. For Landcover task, we implement 1d-CNN with 8 channels as Φ with 400 epochs of training and 350 epochs of annealing. We mainly implement Adam optimizer (Kingma and Ba, 2014), except for the u_{w_u}, q_{w_q}, p_{w_p}, and o_{w_o} which utilize the SGD optimizer (Bottou, 2012) to improve the adversarial training robustness.

F. Examples of $\boldsymbol{X} = [\boldsymbol{X}_v, \boldsymbol{X}_s, \boldsymbol{X}_z]$

Here, we use the annotation $\boldsymbol{X}$ to represent generalized features that may contain all the information about the sample. It is important to note that in reality, there may exist thousands of features for just one sample, including invariant, spurious, and other features that are not related to the prediction task of Y . To fully understand the assumption of , we can consider several examples:

- In DNA expression level prediction task, the regulatory process is complex, involving various transcription factors and epigenetic modifications on millions of basic-pairs (e.g., ATGC). Only a small subset of these factors are directly responsible for the expression level of a gene (e.g., direct cause). Hence, the direct factors can be considered as $\boldsymbol{X}_v$ while the remaining factors can be considered as $\boldsymbol{X}_s$. In addition, there may be other factors to link important transcription factors, measurement noise, sequencing errors, or other irrelevant information for predicting expression that can be considered as $\boldsymbol{X}_z$.
- In search and recommendation scenarios, users' behavior data may include various features such as search queries, browsing history, demographics, and social interactions. However, not all of these features are equally important for predicting user preferences or purchase behaviors. Some features such as the user's age, gender, and occupation may be more relevant than others such as the time of the day or the device used for searching. Hence, the relevant features that direct cause changes in preference can be considered as $\boldsymbol{X}_v$, while the other relevant features can be considered as $\boldsymbol{X}_s$. Moreover, some features may not be relevant at all, such as the user's IP address or the browser type, can be considered as $\boldsymbol{X}_z$.
- In image recognition domains, each image may contain multiple objects, backgrounds, and visual cues. For example, in the image of a cat sitting on a sofa, the cat is the primary object of interest, and the sofa is the background. Other visual cues such as the texture of the floor or the pattern of the curtains may not be relevant to the recognition task. Hence, the features related to the cat and the sofa can be considered as $\boldsymbol{X}_v$ and $\boldsymbol{X}_s$, respectively, while the irrelevant features, such as floor textures, can be considered as $\boldsymbol{X}_z$. Furthermore, in real-world applications, there may be noise, occlusions, or other irrelevant visual information that can also be considered as $\boldsymbol{X}_z$.