Invariance Principle Meets Information Bottleneck for Out-of-Distribution Generalization

Kartik Ahuja[†] **Ethan Caballero**[*†] **Dinghuai Zhang**[*†]

Jean-Christophe Gagnon-Audet[†] **Yoshua Bengio**[†] **Ioannis Mitliagkas**[†]

Irina Rish[†]

Abstract

The invariance principle from causality is at the heart of notable approaches such as invariant risk minimization (IRM) that seek to address out-of-distribution (OOD) generalization failures. Despite the promising theory, invariance principle-based approaches fail in common classification tasks, where invariant (causal) features capture all the information about the label. Are these failures due to the methods failing to capture the invariance? Or is the invariance principle itself insufficient? To answer these questions, we revisit the fundamental assumptions in linear regression tasks, where invariance-based approaches were shown to provably generalize OOD. In contrast to the linear regression tasks, we show that for linear classification tasks we need much stronger restrictions on the distribution shifts, or otherwise OOD generalization is impossible. Furthermore, even with appropriate restrictions on distribution shifts in place, we show that the invariance principle alone is insufficient. We prove that a form of the *information bottleneck* constraint along with invariance helps address key failures when invariant features capture all the information about the label and also retains the existing success when they do not. We propose an approach that incorporates both of these principles and demonstrate its effectiveness in several experiments.

1 Introduction

Recent years have witnessed an explosion of examples showing deep learning models are prone to exploiting shortcuts (spurious features) (Geirhos et al., 2020; Pezeshki et al., 2020) which make them fail to generalize out-of-distribution (OOD). In Beery et al. (2018), a convolutional neural network was trained to classify camels from cows; however, it was found that the model relied on the background color (e.g., green pastures for cows) and not on the properties of the animals (e.g., shape). These examples become very concerning when they occur in real-life applications (e.g., COVID-19 detection (DeGrave et al., 2020)).

To address these out-of-distribution generalization failures, invariant risk minimization (Arjovsky et al., 2019) and several other works were proposed (Ahuja et al., 2020; Pezeshki et al., 2020; Krueger et al., 2020; Robey et al., 2021; Zhang et al., 2021). The invariance principle from causality (Peters et al., 2015; Pearl, 1995) is at the heart of these works. The principle distinguishes predictors that only rely on the causes of the label from those that do not. The optimal predictor that only focuses on the causes is invariant and min-max optimal (Rojas-Carulla et al., 2018; Koyama and Yamaguchi, 2020; Ahuja et al., 2021b) under many distribution shifts but the same is not true for other predictors.

[*]Equal contribution.

[†]Mila - Quebec AI Institute, Université de Montréal. Correspondence to: kartik.ahuja@mila.quebec.

35th Conference on Neural Information Processing Systems (NeurIPS 2021).

Our contributions. Despite the promising theory, invariance principle-based approaches fail in settings (Aubin et al., 2021) where invariant features capture all information about the label contained in the input. A particular example is image classification (e.g., cow vs. camel) (Beery et al., 2018) where the label is a deterministic function of the invariant features (e.g., shape of the animal), and does not depend on the spurious features (e.g., background). To understand such failures, we revisit the fundamental assumptions in linear regression tasks, where invariance-based approaches were shown to provably generalize OOD. We show that, in contrast to the linear regression tasks, OOD generalization is significantly harder for linear classification tasks; we need much stronger restrictions in the form of support overlap assumptions[3] on the distribution shifts, or otherwise it is not possible to guarantee OOD generalization under interventions on variables other than the target class. We then proceed to show that, even under the right assumptions on distribution shifts, the invariance principle is insufficient. However, we establish that *information bottleneck* (IB) constraints (Tishby et al., 2000), together with the invariance principle, provably works in both settings – when invariant features completely capture the information about the label and also when they do not. (Table 1 summarizes our theoretical results presented later). We propose an approach that combines both these principles and demonstrate its effectiveness on linear unit tests (Aubin et al., 2021) and on different real datasets.

Task	Invariant features capture label info	Support overlap invariant features	Support overlap spurious features	**OOD generalization guarantee** (E_{tr} ! E_{all}) ERM	IRM	IB-ERM	IB-IRM	
Linear Classification	Full/Partial	No	Yes/No	Impossible for any algorithm to generalize OOD [Thm2]				
	Full	Yes	No	✗	✗	✓	✓	[Thm3,4]
	Partial	Yes	No	✗	✗	✗	✓	[Appendix]
	Full	Yes	Yes	✓	✓	✓	✓	[Thm3,4]
	Partial	Yes	Yes	✗	✓	✗	✓	
Linear Regression	Full	No	No	✓	✓	✓	✓	
	Partial	No	No	✗	✓	✗	✓	[Thm4]

Table 1: Summary of the new and existing results (Arjovsky et al., 2019; Rosenfeld et al., 2021b). IB-ERM (IRM): information bottleneck - empirical (invariant) risk minimization ERM (IRM).

2 OOD generalization and invariance: background & failures

Background. We consider a supervised training data D gathered from a set of training environments E_{tr}: $D = \{D^e\}_{e \in \mathcal{E}_{tr}}$, where $D^e = \{x_i^e, y_i^e\}_{i=1}^{n^e}$ is the dataset from environment $e \in E_{tr}$ and n^e is the number of instances in environment e. $x_i^e \in \mathbb{R}^d$ and $y_i^e \in Y \subseteq \mathbb{R}^k$ correspond to the input feature value and the label for i^{th} instance respectively. Each (x_i^e, y_i^e) is an i.i.d. draw from P^e, where P^e is the joint distribution of the input feature and the label in environment e. Let X^e be the support of the input feature values in the environment e. The goal of OOD generalization is to use training data D to construct a predictor $f : \mathbb{R}^d \to \mathbb{R}^k$ that performs well across many unseen environments in E_{all}, where $E_{all} \supseteq E_{tr}$. Define the risk of f in environment e as $R^e(f) = \mathbb{E}[\ell(f(X^e), Y^e)]$, where for example ℓ can be 0-1 loss, logistic loss, square loss, $(X^e, Y^e) \sim P^e$, and the expectation $\mathbb{E}$ is w.r.t. P^e. Formally stated, our goal is to use the data from training environments E_{tr} to find $f : \mathbb{R}^d \to Y$ to minimize

$$\min_{f} \max_{e \in \mathcal{E}_{all}} R^e(f). \tag{1}$$

So far we did not state any restrictions on E_{all}. Consider binary classification: without any restrictions on E_{all}, no method can reduce the above objective (ℓ is 0-1 loss) to below one. Suppose a method outputs f^*; if $\exists e \in E_{all} \cap E_{tr}$ with labels based on $1 - f^*$, then it achieves an error of one. Some assumptions on E_{all} are thus necessary. Consider how E_{all} is restricted using invariance for linear regressions (Arjovsky et al., 2019).

Assumption 1. ***Linear regression structural equation model (SEM).*** *In each $e \in E_{all}$*

$$\begin{aligned} Y^e &\leftarrow w_{inv}^* Z_{inv}^e + \epsilon^e; \quad Z_{inv}^e \perp \epsilon^e; \quad \mathbb{E}[\epsilon^e] = 0; \mathbb{E}\,|\epsilon^e|^2 \leq \sigma_{sup}^2 \\ X^e &\leftarrow S(Z_{inv}^e, Z_{spu}^e) \end{aligned} \tag{2}$$

where $w_{inv}^* \in \mathbb{R}^m$, $Z_{inv}^e \in \mathbb{R}^m$, $Z_{spu} \in \mathbb{R}^o$, $S \in \mathbb{R}^{d \times (m+o)}$, S is invertible ($m + o = d$). We focus on invertible S but several results extend to non-invertible S as well (see Appendix).

[3]Support is the region where the probability density for continuous random variables (probability mass function for discrete random variables) is positive. Support overlap refers to the setting where train and test distribution maybe different but share the same support. We formally define this later in Assumption 5.

Assumption 1 states how Y^e and X^e are generated from latent invariant features Z^e_{inv}[4], latent spurious features Z^e_{spu} and noise ^{e}. The *relationship between label and invariant features is invariant, i.e.,* w^*_{inv} *is fixed* across all environments. However, the distributions of Z^e_{inv}, Z^e_{spu}, and ^{e} are allowed to change arbitrarily across all the environments. Suppose S is identity. If we regress only on the invariant features Z^e_{inv}, then the optimal solution is w^*_{inv}, which is independent of the environment, and the error it achieves is bounded above by the variance of ^{e} ($^2_{\text{sup}}$). If we regress on the entire Z^e and the optimal predictor places a non-zero weight on Z^e_{spu} (e.g., Z^e_{spu} Y^e + ^{e}), then this predictor fails to solve equation (1) ($9\ e\ 2\ E_{all}$, Z^e_{spu} ! 1 , error ! 1 , see Appendix for details). Also, not only regressing on Z^e_{inv} is better than on Z^e, it can be shown that it is optimal, i.e., it solves equation (1) under Assumption 1 and achieves a value of $^2_{\text{sup}}$ for the objective in equation (1).

Invariant predictor. Define a linear representation map : $\mathbb{R}^{r\times d}$ (that transforms X^e as (X^e)) and define a linear classifier $w : \mathbb{R}^{k\times r}$ (that operates on the representation w (X^e)). We want to search for representations such that $\mathbb{E}[Y^e j$ (X^e)] is invariant (in Assumption 1 if (X^e) = Z^e_{inv}, then $\mathbb{E}[Y^e j$ (X^e)] is invariant). We say that a data representation elicits an invariant predictor w across the set of training environments E_{tr} if there is a predictor w that simultaneously achieves the minimum risk, i.e., $w\ 2\ \arg\min_w R^e(w$); $8e\ 2\ E_{tr}$. The main objective of IRM is stated as

$$\min_{w\in\mathbb{R}^{k\ r};\ \in\mathbb{R}^{r\ d}} \frac{1}{jE_{tr}j}\sum_{e\in\mathcal{E}_{tr}} R^e(w\ \ ) \quad \text{s.t. } w\ 2\ \arg\min_{w\in\mathbb{R}^{k\ r}} R^e(w\ \ );\ 8e\ 2\ E_{tr}: \tag{3}$$

Observe that if we drop the constraints in the above which search only over invariant predictors, then we get the standard empirical risk minimization (ERM) (Vapnik, 1992) (assuming all the training environments occur with equal probability). In all our theorems, we use 0-1 loss for binary classification $Y = f0; 1g$ and square loss for regression $Y = \mathbb{R}$. For binary classification, the output of the predictor is given as $\mathbb{I}(w$ (X^e)), where $\mathbb{I}(\ )$ is the indicator function that takes 1 if the input is 0 and 0 otherwise, and the risk is $R^e(w\ \ ) = \mathbb{E}\ j\mathbb{I}(w$ (X^e)) $Y^e j$. For regression, the output of the predictor is w (X^e) and the corresponding risk is $R^e(w\ \ ) = \mathbb{E}\ (w$ (X^e) $Y^e)^2$. We now present the main OOD generalization result from Arjovsky et al. (2019) for linear regressions.

Theorem 1. *(Informal) If Assumption 1 is satisfied,* $\text{Rank}[\ ] > 0$, $jE_{tr}j > 2d$, *and* E_{tr} *lie in a linear general position (a mild condition on the data in* E_{tr}*, defined in the Appendix), then each solution to equation* (3) *achieves OOD generalization (solves equation* (1)*,* @ $e\ 2\ E_{all}$ *with risk* > $^2_{\text{sup}}$*).*

Despite the above guarantees, IRM has been shown to fail in several cases including linear SEMs in (Aubin et al., 2021). We take a closer look at these failures next.

Understanding the failures: fully informative invariant features vs. partially informative invariant features (FIIF vs. PIIF). We define properties salient to the datasets/SEMs used in the OOD generalization literature. Each $e\ 2\ E_{all}$, the distribution ($X^e; Y^e$) P^e satisfies the following properties. a) 9 a map * (linear or not), which we call an *invariant feature map*, such that $\mathbb{E}\ Y^e$ * X^e is the same for all $e\ 2\ E_{all}$ and Y^e $\mathcal{G}$ $^*(X^e)$. These conditions ensure * maps to features that have a finite predictive power and have the same optimal predictor across E_{all}. For the SEM in Assumption 1, * maps to Z^e_{inv}. b) 9 a map * (linear or not), which we call *spurious feature map*, such that $\mathbb{E}\ Y^e$ * X^e is not the same for all $e\ 2\ E_{all}$ and Y^e $\mathcal{G}$ $^*(X^e)$ for some environments. * often creates a hindrance in learning predictors that only rely on *. Note that * should not be a transformation of some *. For the SEM in Assumption 1, suppose Z^e_{spu} is anti-causally related to Y^e, then * maps to Z^e_{spu} (See Appendix for an example).

In the colored MNIST (CMNIST) dataset (Arjovsky et al., 2019), the digits are colored in such a way that in the training domain, color is highly predictive of the digit label but this correlation being spurious breaks down at test time. Suppose the invariant feature map * extracts the uncolored digit and the spurious feature map * extracts the background color. Ahuja et al. (2021b) studied two variations of the colored MNIST dataset, which differed in the way final labels are generated from original MNIST labels (corrupted with noise or not). They showed that the IRM exhibits good OOD generalization (50% improvement over ERM) in anti-causal-CMNIST (AC-CMNIST, original data from Arjovsky et al. (2019)) but is no different from ERM and fails in covariate shift-CMNIST (CS-CMNIST). In AC-CMNIST, the invariant features $^*(X^e)$ (uncolored digit) are *partially informative* about the label, i.e., Y $\mathcal{G}$ $X^e j$ $^*(X^e)$, and color contains information about label not contained

[4]In many examples in the literature, invariant features are causal, but not always (Rosenfeld et al., 2021b).

Fully informative invariant features (FIIF)	**Partially informative invariant features (PIIF)**
$8e \ 2 \ E_{all}; Y^e \ ? \ X^e j \ ^*(X^e)$	$9 \ e \ 2 \ E_{all} \ Y^e \ 6? \ X^e j \ ^*(X^e)$
Task: classification	**Task: classification or regression**
Example 2/2S, CS-CMNIST	Example 1/1S, Example 3/3S, AC-CMNIST
SEM in Assumption 2	SEM in Rosenfeld et al. (2021b)
ERM and IRM fail	**ERM fails, IRM succeeds sometimes**
Theorem 3,4 (This paper)	Theorem 9, 5.1 (Arjovsky et al., 2019; Rosenfeld et al., 2021b)

Table 2: Categorization of OOD evaluation datasets and SEMs. Example 1/1S, 2/2S, 3/3S from (Aubin et al., 2021), AC-CMNIST(Arjovsky et al., 2019), CS-CMNIST(Ahuja et al., 2021b).

in the uncolored digit. On the other hand in CS-CMNIST, invariant features are *fully informative* about the label, i.e., $Y \ ? \ X^e j \ ^*(X^e)$, i.e., they contains all the information about the label that is contained in input X^e. Most human labelled datasets have fully informative invariant features; the labels (digit value) only depend on the invariant features (uncolored digit) and spurious features (color of the digit) do not affect the label. [5] In the rare case, when the humans are asked to label images in which the object being labelled itself is blurred, humans can rely on spurious features such as the background making such a data representative of PIIF setting. In Table 2, we divide the different datasets used in the literature based on informativeness of the invariant features. We observe that when the invariant features are fully informative, both IRM and ERM fail but only in classification tasks and not in regression tasks (Ahuja et al., 2021b); this is consistent with the linear regression result in Theorem 1, where IRM succeeds regardless of whether $Y^e \ ? \ X^e j Z^e_{\text{inv}}$ holds or not. Motivated by this observation, we take a closer look at the classification tasks where invariant features are fully informative.

3 OOD generalization theory for linear classification tasks

A two-dimensional example with fully informative invariant features. We start with a 2D classification example (based on Nagarajan et al. (2021)), which can be understood as a simplified version of the CS-CMNIST dataset (Ahuja et al., 2021b), Example 2/2S of Aubin et al. (2021), where both IRM and ERM fail. The example goes as follows. In each training environment $e \ 2 \ E_{tr}$

$$\begin{aligned} Y^e \quad & \mathsf{I} \ X^e_{\text{inv}} \quad \frac{1}{2} \ ; \text{ where } X^e_{\text{inv}} \ 2 \ f0; 1g \text{ is Bernoulli } \frac{1}{2} \ ; \\ X^e_{\text{spu}} \quad & X^e_{\text{inv}} \quad W^e; \text{ where } W^e \ 2 \ f0; 1g \text{ is Bernoulli } 1 \quad p^e \quad \text{with selection bias } p^e > \frac{1}{2}; \end{aligned} \tag{4}$$

where $\mathsf{Bernoulli}(a)$ takes value 1 with probability a and 0 otherwise. Each training environment is characterized by the probability p^e. Following Assumption 1, we assume that the labelling function does not change from E_{tr} to E_{all}, thus the relation between the label and the invariant features does not change. Assume that the distribution of X^e_{inv} and X^e_{spu} can change arbitrarily. See Figure 1a) for a pictorial representation of this example illustrating the gist of the problem: there are many classifiers with the same error on E_{tr} while only the one identical to the labelling function $\mathsf{I}(X^e_{\text{inv}} \quad \frac{1}{2})$ generalizes correctly OOD. Define a classifier $\mathsf{I}(w_{\text{inv}} x_{\text{inv}} + w_{\text{spu}} x_{\text{spu}} \quad \frac{1}{2}(w_{\text{inv}} + w_{\text{spu}}))$. Define a set of classifiers $S = f(w_{\text{inv}}; w_{\text{spu}})$ s.t. $w_{\text{inv}} > j w_{\text{spu}} j g$. Observe that all the classifiers in S achieve a zero classification error on the training environments. However, only classifiers for which $w_{\text{spu}} = 0$ solve the OOD generalization (eq. (1)). With as the identity, it can be shown that all the classifiers S form an invariant predictor (satisfy the constraint in equation (3) over all the training environments when ℓ is the 0-1 loss). Observe that increasing the number of training environments to infinity does not address the problem, unlike with the linear regression result discussed in Theorem 1 (Arjovsky et al., 2019), where it was shown that if the number of environments increases linearly in the dimension of the data, then the solution to IRM also solves the OOD generalization (eq. (1)). [6] We use the above example to construct general SEMs for linear classification when the invariant features are fully informative. We follow the structure of the SEM from Assumption 1 in our construction.

[5] The deterministic labelling case was referred as realizable problems in (Arjovsky et al., 2019).

[6] Please note that this example illustrates certain important facets in a very simple fashion; only in this example a max-margin classifier can solve the problem but not in general. (Further explanation in the Appendix).

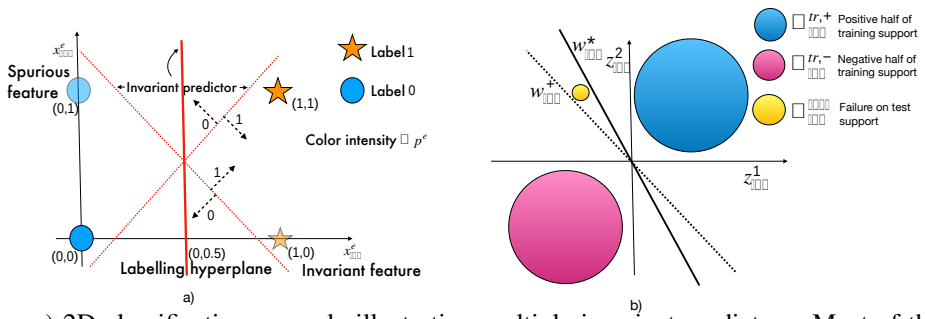


Figure 1: a) 2D classification example illustrating multiple invariant predictors: Most of these predictors rely on spurious features and each of them achieve zero error across all E_{tr}, b) illustration of the impossibility result. If latent invariant features in the training environments are separable, then there are multiple equally good candidates that could have generated the data, and the algorithm cannot distinguish between these.

Assumption 2. ***Linear classification structural equation model (FIIF).*** *In each* $e \in E_{all}$

$$\begin{aligned} Y^e &:= \mathbb{I}(w^*_{\text{inv}} \cdot Z^e_{\text{inv}}) \oplus N^e; \quad N^e \sim \text{Bernoulli}(q); q < \frac{1}{2}; \quad N^e \perp (Z^e_{\text{inv}}, Z^e_{\text{spu}}); \\ X^e &:= S \cdot (Z^e_{\text{inv}}, Z^e_{\text{spu}}); \end{aligned} \tag{5}$$

where $w^*_{\text{inv}} \in \mathbb{R}^m$ *with* $\|w^*_{\text{inv}}\| = 1$ *is the labelling hyperplane,* $Z^e_{\text{inv}} \in \mathbb{R}^m$, $Z^e_{\text{spu}} \in \mathbb{R}^o$, N^e *is binary noise with identical distribution across environments,* $\oplus$ *is the XOR operator,* S *is invertible.*

If noise level q is zero, then the above SEM covers linearly separable problems. See Figure 2a) for the directed acyclic graph (DAG) corresponding to this SEM. From the DAG observe that $Y^e \perp X^e | Z^e_{\text{inv}}$, which implies that the invariant features are fully informative. Contrast this with a DAG that follows Assumption 1 shown in Figure 2b), where $Y^e \not\perp X^e | Z^e_{\text{inv}}$ and thus the invariant features are not fully informative. If E_{all} follows the SEM in Assumption 2 and suppose the distribution of Z^e_{inv}, Z^e_{spu} can change arbitrarily, then it can be shown that only a classifier identical to the labelling function $\mathbb{I}(w^*_{\text{inv}} \cdot Z^e_{\text{inv}})$ can solve the OOD generalization (eq. (1)); such a classifier achieves an error of q (noise level) in all the environments. As a result, if for a classifier we can find $e \in E_{all}$ that follows Assumption 2 where the error is greater than q, then such a classifier does not solve equation (1). Now we ask – what are the minimal conditions on training environments E_{tr} to achieve OOD generalization when E_{all} follow Assumption 2? To achieve OOD generalization for linear regressions, in Theorem 1, it was required that the number of training environments grows linearly in the dimension of the data. However, there was no restriction on the support of the latent invariant and latent spurious features, and they were allowed to change arbitrarily from train to test (for further discussion on this, see the Appendix). Can we continue to work with similar assumptions for the SEM in Assumption 2 and solve the OOD generalization (eq. (1))? We state some assumptions and notations to answer that. Define the support of the invariant (spurious) features Z^e_{inv} (Z^e_{spu}) in environment e as Z^e_{inv} (Z^e_{spu}).

Assumption 3. ***Bounded invariant features.*** $\bigcap_{e \in \mathcal{E}_{tr}} Z^e_{\text{inv}}$ *is a bounded set.*[7]

Assumption 4. ***Bounded spurious features.*** $\bigcap_{e \in \mathcal{E}_{tr}} Z^e_{\text{spu}}$ *is a bounded set.*

Assumption 5. ***Invariant feature support overlap.*** $\forall e \in E_{all}; Z^e_{\text{inv}} \quad \bigcap_{e^0 \in \mathcal{E}_{tr}} Z^{e^0}_{\text{inv}}$

Assumption 6. ***Spurious feature support overlap.*** $\forall e \in E_{all}; Z^e_{\text{spu}} \quad \bigcap_{e^0 \in \mathcal{E}_{tr}} Z^{e^0}_{\text{spu}}$

Assumption 5 (6) states that the support of the invariant (spurious) features for unseen environments is the same as the union of the support over the training environments. It is important to note that support overlap does not imply that the distribution over the invariant features does not change. We now define a margin that measures how much the is training support of invariant features Z^e_{inv} separated by the labelling hyperplane w^*_{inv}. Define $\text{Inv-Margin} = \min_{z \in \cup_{e \in E_{tr}} Z^e_{\text{inv}}} \text{sgn}\left(w^*_{\text{inv}} \cdot z\right) \|w^*_{\text{inv}} \cdot z\|$. This margin only coincides with the standard margin in support vector machines when the noise level q is 0 (linearly separable) and S is identity. If $\text{Inv-Margin} > 0$, then the labelling hyperplane w^*_{inv} separates the support into two halves (see Figure 1b)).

[7] A set Z is bounded if $\exists M < \infty$ such that $\forall z \in Z; \|z\| \leq M$.

Assumption 7. ***Strictly separable invariant features.*** Inv-Margin > 0.

Next, we show the importance of support overlap for invariant features.

Theorem 2. ***Impossibility of guaranteed OOD generalization for linear classification.*** *Suppose each $e \in E_{all}$ follows Assumption 2. If for all the training environments E_{tr}, the latent invariant features are bounded and strictly separable, i.e., Assumption 3 and 7 hold, then every deterministic algorithm fails to solve the OOD generalization (eq.* (1)*), i.e., for the output of every algorithm $\exists e \in E_{all}$ in which the error exceeds the minimum required value γ (noise level).*

The proofs to all the theorems are in the Appendix. We provide a high-level intuiton as to why invariant feature support overlap is crucial to the impossibility result. In Figure 1b), we show that if the support of latent invariant features are strictly separated by the labelling hyperplane w^*_{inv}, then we can find another valid hyperplane w^+_{inv} that is equally likely to have generated the same data. There is no algorithm that can distinguish between w^*_{inv} and w^+_{inv}. As a result, if we use data from the region where the hyperplanes disagree (yellow region Figure 1b)), then the algorithm fails.

Significance of Theorem 2. We showed that without the support overlap assumption on the invariant features, OOD generalization is impossible for linear classification tasks. This is in contrast to linear regression in Theorem 1 (Arjovsky et al., 2019), where even in the absence of the support overlap assumption, guaranteed OOD generalization was possible. Applying the above Theorem 2 to the 2D case (eq. (4)) implies that we cannot assume that the support of invariant latent features can change, or else that case is also impossible to solve.

Next, we ask what further assumptions are minimally needed to be able to solve the OOD generalization (eq. (1)). Each classifier can be written as $w \circ X^e = w \circ S(Z^e_{\text{inv}}, Z^e_{\text{spu}}) = w_{\text{inv}} \circ Z^e_{\text{inv}} + w_{\text{spu}} Z^e_{\text{spu}}$. If $w_{\text{spu}} \neq 0$, then the classifier w is said to rely on spurious features.

Theorem 3. ***Sufficiency and Insufficiency of ERM and IRM.*** *Suppose each $e \in E_{all}$ follows Assumption 2. Assume that a) the invariant features are strictly separable, bounded, and satisfy support overlap, b) the spurious features are bounded (Assumptions 3-5, 7 hold).*

Sufficiency: *If the spurious features satisfy support overlap (Assumption 6 holds), then both ERM and IRM solve the OOD generalization problem (eq.* (1)*). Also, there exist solutions to ERM and IRM solutions that rely on the spurious features and still achieve OOD generalization.*

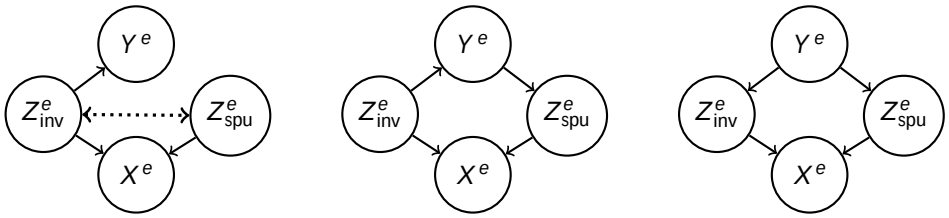
Insufficiency: *If spurious features do not satisfy support overlap, then both ERM and IRM fail at solving the OOD generalization problem (eq.* (1)*). Also, there exist no such classifiers that rely on spurious features and also achieve OOD generalization.*

Significance of Theorem 3. From the first part, we learn that if the support overlap is satisfied for both the invariant features and the spurious features, then either ERM or IRM can solve the OOD generalization (eq. (1)). Interestingly, in this case we can have classifiers that rely on the spurious features and yet solve the OOD generalization (eq. (1)). For the 2D case (eq. (4)) this case implies that the entire set S solves the OOD generalization (eq. (1)). From the second part, we learn that if support overlap holds for invariant features but not for spurious features, then the ideal OOD optimal predictors rely only on the invariant features. In this case, methods like ERM and IRM continue to rely on spurious features and fail at OOD generalization. For the above 2D case (eq. (4)) this implies that only the predictors that rely only on X^e_{inv} in the set S solve the OOD generalization (eq. (1)).

To summarize, we looked at SEMs for classification tasks when invariant features are fully informative, and find that the support overlap assumption over invariant features is necessary. Even in the presence of support overlap for invariant features, we showed that ERM and IRM can easily fail if the support overlap is violated for spurious features. This raises a natural question – Can we even solve the case with the support overlap assumption only on the invariant features? We will now show that the information bottleneck principle can help tackle these cases.

4 Information bottleneck principle meets invariance principle

Why the information bottleneck? The information bottleneck principle prescribes to learn a representation that compresses the input X as much as possible while preserving all the relevant information about the target label Y (Tishby et al., 2000). Mutual information $I(X; \ (X))$ is used to measure information compression. If representation (X) is a deterministic transformation of X, then in principle we can use the entropy of (X) to measure compression (Kirsch et al., 2020). Let



(a) FIIF (this work) (b) PIIF (Arjovsky et al., 2019) (c) PIIF (Rosenfeld et al., 2021b)

Figure 2: Comparison of the DAG from Assumption 2 (fully informative invariant features) vs. DAGs from Rosenfeld et al. (2021b); Arjovsky et al. (2019) (partially informative invariant features).

us revisit the 2D case (eq. (4)) and apply this principle to it. Following the second part of Theorem 3, where ERM and IRM failed, assume that invariant features satisfy the support overlap assumption, but make no such assumption for the spurious features. Consider three choices for : identity (selects both features), selects invariant feature only, selects spurious feature only. The entropy of $H(\ (X^e))$ when is the identity is $H(p^e) + \log(2)$, where $H(p^e)$ is the Shannon entropy in $\text{Bernoulli}(p^e)$. If selects the invariant/spurious features only, then $H(\ (X^e)) = \log(2)$. Among all three choices, the one that has the least entropy and also achieves zero error is the representation that focuses on the invariant feature. We could find the OOD optimal predictor in this example just by using information bottleneck. Does it mean the invariance principle isn't needed? We answer this next.

Why invariance? Consider a simple classification SEM. In each $e \in E_{tr}$, Y^e $X^{1;e}_{\text{inv}}$ $X^{2;e}_{\text{inv}}$ N^e and X^e_{spu} Y^e V^e, where all the random variables involved are binary valued, noise $N^e; V^e$ are Bernoulli with parameters q (identical across E_{tr}), c^e (varies across E_{tr}) respectively. If $c^e < q$, then in E_{tr} predictions based on X^e_{spu} are better than predictions based on $X^{1;e}_{\text{inv}}; X^{2;e}_{\text{inv}}$. If both $X^{1;e}_{\text{inv}}; X^{2;e}_{\text{inv}}$ are uniform Bernoulli, then these features have a higher entropy than X^e_{spu}. In this case, the information bottleneck would bar using $X^{1;e}_{\text{inv}}; X^{2;e}_{\text{inv}}$. Instead, we want the model to focus on $X^{1;e}_{\text{inv}}$, $X^{2;e}_{\text{inv}}$ and not on X^e_{spu}. Invariance constraints encourage the model to focus on $X^{1;e}_{\text{inv}}$, $X^{2;e}_{\text{inv}}$. In this example, observe that invariant features are partially informative unlike the 2D case (eq. (4)).

Why invariance and information bottleneck? We have illustrated through simple examples when the information bottleneck is needed but not invariance and vice-versa. We now provide a simple example where both these constraints are needed at the same time. This example combines the 2D case (eq. (4)) and the example we highlighted in the paragraph above: Y^e X^e_{inv} N^e, $X^{1;e}_{\text{spu}}$ X^e_{inv} W^e, and $X^{2;e}_{\text{spu}}$ Y^e V^e. In this case, the invariance constraint does not allow representations that use $X^{2;e}_{\text{spu}}$ but does not prohibit representations that rely on $X^{1;e}_{\text{spu}}$. However, information bottleneck constraints on top ensure that representations that only use X^e_{inv} are used. We now describe an objective [8] that combines both these principles:

$$\min_{w;} \sum_{e\in\mathcal{E}_{tr}} h^e\ w \qquad \text{s.t.} \frac{1}{|E_{tr}|}\sum_{e\in\mathcal{E}_{tr}} R^e\ w \qquad r^{\text{th}};\ w \in \arg\min_{w\in\mathbb{R}^k\ r} R^e(w\quad);\forall e \in E_{tr}; \quad (6)$$

where h^e in the above is a lower bounded differential entropy defined below and r^{th} is the threshold on the average risk. Typical information bottleneck based optimization in neural networks involves minimization of the entropy of the representation output from a certain hidden layer. For both analytical convenience and also because the above setup is a linear model, we work with the simplest form of bottleneck which directly minimizes the entropy of the output layer. Recall the definition of differential entropy of a random variable X, $h(X) = \quad \mathbb{E}_X[\log d\mathbb{P}_X]$ and $d\mathbb{P}_X$ is the Radon-Nikodym derivative of $\mathbb{P}_X$ with respect to Lebesgue measure. Because in general differential entropy has no lower bound, we add a small independent noise term (Kirsch et al., 2020) to the classifier to ensure that the entropy is bounded below. We call the above optimization information bottleneck based invariant risk minimization (IB-IRM). In summary, *among all the highly predictive invariant predictors we pick the ones that have the least entropy*. If we drop the invariance constraint from the above optimization, we get information bottleneck based empirical risk minimization (IB-ERM). In the above formulation and following result, we assume that X^e are continuous random variables; the results continue to hold for discrete X^e as well (See Appendix for details).

Theorem 4. ***IB-IRM and IB-ERM vs. IRM and ERM***

[8] Results extend to alternate objective with information bottleneck constraints and average risk as objective.

Fully informative invariant features (FIIF). *Suppose each $e \in E_{all}$ follows Assumption 2. Assume that the invariant features are strictly separable, bounded, and satisfy support overlap (Assumptions 3,5 and 7 hold). Also, for each $e \in E_{tr}$ Z^e_{spu} $AZ^e_{inv} + W^e$, where $A \in \mathbb{R}^{o \times m}$, $W^e \in \mathbb{R}^o$ is continuous, bounded, and zero mean noise. Each solution to IB-IRM (eq.* (6)*, with ℓ as 0-1 loss, and $r^{th} = q$), and IB-ERM solves the OOD generalization (eq.* (1)*) but ERM and IRM (eq.*(3)*) fail.*

Partially informative invariant features (PIIF). *Suppose each $e \in E_{all}$ follows Assumption 1 and $\exists e \in E_{tr}$ such that $\mathbb{E}[{}^e Z^e_{spu}] \neq 0$. If $|E_{tr}| > 2d$ and the set E_{tr} lies in a linear general position (a mild condition defined in the Appendix), then each solution to IB-IRM (eq.* (6)*, with ℓ as square loss, $^2 < r^{th}$ ^2_Y, where ^2_Y and ^{2} are the variance in the label and noise across E_{tr}) and IRM (eq.*(3)*) solves OOD generalization (eq.* (1)*) but IB-ERM and ERM fail.*

Significance of Theorem 4 and remarks. In the first part (FIIF), IB-ERM and IB-IRM succeed without assuming support overlap for the spurious features, which was crucial for success of ERM and IRM in Theorem 3. This establishes that support overlap of spurious features is not a necessary condition. Observe that when invariant features are fully informative, IB-ERM and IB-IRM succeed, but when invariant features are partially informative IB-IRM and IRM succeed. In real data settings, we do not know if the invariant features are fully or partially informative. Since IB-IRM is the only common winner in both the settings, it would be pragmatic to use it in the absence of domain knowledge about the informativeness of the invariant features. In the paragraph preceding the objective in equation (6), we discussed examples where both the IB and IRM constraints were needed at the same time. In the Appendix, we generalize that example and show that if we change the assumptions in linear classification SEM in Assumption 2 such that the invariant features are partially informative, then we see the joint benefit of IB and IRM constraints. At this point, it is also worth pointing to a result in Rosenfeld et al. (2021b), which focused on linear classification SEMs (DAG shown in Figure 2c) with partially informative invariant features. Under the assumption of complete support overlap for spurious and invariant features, authors showed IRM succeeds.

4.1 Proposed approach

We take the three terms from the optimization in equation (6) and create a weighted combination as

$$\sum_e R^e(\ ) + k\, r_{w;w=1:0} R^e(w\ )\|^2 + h^e(\ ) \qquad \sum_e R^e(\ ) + k\, r_{w;w=1:0} R^e(w\ )\|^2 + h(\ ) :$$

In the LHS above, the first term corresponds to the risks across environments, the second term approximates invariance constraint (follows the IRMv1 objective (Arjovsky et al., 2019)), and the third term is the entropy of the classifier in each environment. In the RHS, $h(\ )$ is the entropy of unconditional on the environment (the entropy on the left-hand side is entropy conditional on the environment assuming all the environments are equally likely). Optimizing over differential entropy is not easy, and thus we resort to minimizing an upper bound of it (Kirsch et al., 2020). We use the standard result that among all continuous random variables with the same variance, Gaussian has the maximum differential entropy. Since the entropy of Gaussian increases with its variance, we use the variance of instead of the differential entropy (For further details, see the Appendix). Our final objective is given as

$$\sum_e R^e(\ ) + k\, r_{w;w=1:0} R^e(w\ )\|^2 + \mathrm{Var}(\ ) : \quad (7)$$

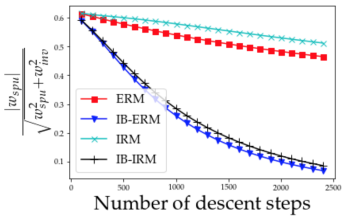


Figure 3: Comparing convergence of $\frac{|w_{spu}|}{\sqrt{w^2_{spu}+w^2_{inv}}}$ (metric from Nagarajan et al. (2021)) for average selection bias $p = 0.9$.

On the behavior of gradient descent with and without information bottleneck. In the entire discussion so far, we have focused on ensuring that the set of optimal solutions to the desired objective (IB-IRM, IB-ERM, etc.) correspond to the solutions of the OOD generalization problem (eq. (1)). In some simple cases, such as the 2D case (eq. (4)), it can be shown that gradient descent is biased towards selecting the ideal classifier (Soudry et al., 2018; Nagarajan et al., 2021). Even though gradient descent can eventually learn the ideal classifier that only relies on the invariant features, training is frustratingly slow as was shown by Nagarajan et al. (2021). In the next theorem, we characterize the impact of using IB penalty ($\mathrm{Var}(\ )$) in the 2D example (eq. (4)). We compare the methods in terms of $|\frac{w_{spu}(t)}{w_{inv}(t)}|$, which was the metric used in Nagarajan et al. (2021); $w_{spu}(t)$ and $w_{inv}(t)$ are the weights for the spurious feature and the invariant feature at time t of training (assuming training happens with continuous time gradient descent).

Theorem 5. ***Impact of IB on learning speed.*** *Suppose each $e \in E_{tr}$ follows the 2D case from equation (4). Set $= 0$, > 0 in equation (7) to get the IB-ERM objective with ℓ as exponential loss. Continuous-time gradient descent on this IB-ERM objective achieves $|\frac{w_{spu}(t)}{w_{inv}(t)}|$ in time less than $\frac{W_0(\frac{1}{2})}{2(1-p)}$ ($W_0(\cdot)$ denotes the principal branch of the Lambert W function), while in the same time the ratio for ERM $|\frac{w_{spu}(t)}{w_{inv}(t)}| \geq \ln(\frac{1+2p}{3-2p}) = \ln\left(1 + \frac{W_0(\frac{1}{2})}{2(1-p)}\right)$, where $p = \frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}} p^e$.*

$|\frac{w_{spu}(t)}{w_{inv}(t)}|$ converges to zero for both methods, but it converges much faster for IB-ERM (for $p = 0.9$; $= 0.001$; $= 0.58$, the ratio for IB-ERM is $|\frac{w_{spu}(t)}{w_{inv}(t)}| \geq 0.001$ and ratio for ERM is $|\frac{w_{spu}(t)}{w_{inv}(t)}| \geq 0.09$). In the above theorem, we analyzed the impact of information bottleneck only. The convergence analysis for both the penalties jointly comes with its own challenges, and we hope to explore this in future work. However, we carried out experiments with gradient descent on all the objectives for the 2D example (eq. (4)). See Figure 3 for the comparisons.

5 Experiments

Methods, datasets & metrics. We compare our approaches – information bottleneck based ERM (IB-ERM) and information bottleneck based IRM (IB-IRM) with ERM and IRM. We also compare with an Oracle model trained on data where spurious features are permuted to remove spurious correlations. We use all the datasets in Table 2, Terra Incognita dataset (Beery et al., 2018), and COCO (Ahmed et al., 2021). We follow the same protocol for tuning hyperparameters from Aubin et al. (2021); Arjovsky et al. (2019) for their respective datasets (see the Appendix for more details). As is reported in literature, for Example 2/2S, Example 3/3S we use classification error and for AC-CMNIST, CS-CMNIST, Terra Incognita, and COCO we use accuracy. For Example 1/1S, we use mean square error (MSE). The code for experiments can be found at `https://github.com/ahujak/IB-IRM`.

Summary of results. In Table 3, we provide a comparison of methods for different examples in linear unit tests (Aubin et al., 2021) for three and six training environments. In Table 4, we provide a comparison of the methods for different CMNIST datasets, Terra Incognita and COCO dataset. Based on our Theorem 4, we do not expect ERM and IB-ERM to do well on Example 1/1S, Example 3/3S and AC-CMNIST as these datasets fall in the PIIF category, i.e, the invariant features are partially informative. On these examples, we find that IRM and IB-IRM do better than ERM and IB-ERM (for Example 3/3S when there are three environments all methods perform poorly). Based on our Theorem 4, we do not expect IRM and ERM to do well on Example 2/2S, CS-CMNIST, Terra Incognita and COCO dataset,[9] as these datasets fall in the FIIF category, i.e., the invariant features are fully informative. On these FIIF examples, we find that IB-ERM always performs well (close to oracle), and in some cases IB-IRM also performs well. Our experiments confirm that IB penalty has a crucial role to play in FIIF settings and IRMv1 penalty has a crucial role to play in PIIF settings (to further this claim, we provide an ablation study in the Appendix). On Example 1/1S, AC-CMNIST, we find that IB-IRM is able to extract the benefit of IRMv1 penalty. On CS-CMNIST and Example 2/2S we find that IB-IRM is able to extract the benefit of IB penalty. In settings such as COCO dataset, where IB-IRM does not perform as well as IB-ERM, better hyperparameter tuning strategies should be able to help IB-IRM adapt and put a higher weight on IB penalty. Overall, we can conclude that IB-ERM improves over ERM (significantly in FIIF and marginally in PIIF settings), and IB-IRM improves over IRM (improves in FIIF settings and retains advantages in PIIF settings).

Remark. As we move from three to six environments, we observe that MSE in Example 1/1S exhibits a larger variance. This is because of the way data is generated, the new environments that are sampled have labels that have a higher noise level (we follow the same procedure as in Aubin et al. (2021)).

6 Extensions, limitations, and future work

Extension to non-linear models and multi-class classification. In this work our theoretical analysis focused on linear models. Consider the map $X \quad S(Z_{inv}; Z_{spu})$ in Assumption 2. Suppose S is non-linear and bijective. We can divide the learning task into two parts a) invert S to obtain $Z_{inv}; Z_{spu}$ and b) learn a linear model that only relies on the invariant features Z_{inv} to predict the label Y. For

[9] We place Terra Incognita and COCO dataset in the FIIF assuming that the humans who labeled the images did not need to rely on unreliable/spurious features such as background to generate the labels.

	#Envs	ERM		IB-ERM		IRM		IB-IRM		Oracle	
Example1	3	13.36	1.49	12.96	1.30	11.15	0.71	11.68	0.90	10.42	0.16
Example1s	3	13.33	1.49	12.92	1.30	11.07	0.68	11.74	1.03	10.45	0.19
Example2	3	0.42	0.01	0.00	0.00	0.45	0.00	0.00	0.00	0.00	0.00
Example2s	3	0.45	0.01	0.00	0.01	0.45	0.01	0.06	0.12	0.00	0.00
Example3	3	0.48	0.07	0.49	0.06	0.48	0.07	0.48	0.07	0.01	0.00
Example3s	3	0.49	0.06	0.49	0.06	0.49	0.07	0.49	0.07	0.01	0.00
Example1	6	33.74	60.18	32.03	57.05	23.04	40.64	25.66	45.96	22.21	39.25
Example1s	6	33.62	59.80	31.92	56.70	22.92	40.60	25.60	45.62	22.13	38.93
Example2	6	0.37	0.06	0.02	0.05	0.46	0.01	0.43	0.11	0.00	0.00
Example2s	6	0.46	0.01	0.02	0.06	0.46	0.01	0.45	0.10	0.00	0.00
Example3	6	0.33	0.18	0.26	0.20	0.14	0.18	0.19	0.19	0.01	0.00
Example3s	6	0.36	0.19	0.27	0.20	0.14	0.18	0.19	0.19	0.01	0.00

Table 3: Comparisons on linear unit tests in terms of mean square error (regression) and classification error (classification). "#Envs" means the number of training environments.

	ERM		IB-ERM		IRM		IB-IRM	
CS-CMNIST	60.27	1.21	71.80	0.69	61.49	1.45	71.79	0.70
AC-CMNIST	16.84	0.82	50.24	0.47	66.98	1.65	67.67	1.78
Terra Incognita	49.80	4.40	56.40	2.10	54.60	1.30	54.10	2.00
COCO	22.70	1.04	31.66	2.39	18.47	10.20	25.10	1.03

Table 4: Classification accuracy percentage on colored MNISTs, Terra Incognita and COCO dataset.

part b), we can rely on the approaches proposed in this work. For part a), we need to leverage advancements in the field of non-linear ICA (Khemakhem et al., 2020). The current state-of-the-art to solve part a) requires strong structural assumptions on the dependence between all the components of $Z_{\mathsf{inv}}; Z_{\mathsf{spu}}$ (Lu et al., 2021). Therefore, solving part a) and part b) in conjunction with minimal assumptions forms an exciting future work. In the entire work, the discussion was focused on binary classification tasks and regression tasks. For multi-class classification settings, we consider natural extension of the SEM in Assumption 2 (See the Appendix) and our main results continue to hold.

On the choice for IB penalty and IRMv1 penalty. We use the approximation for entropy (in equation (7)) described in Kirsch et al. (2020). The approximation (even though an upper bound) serves as an effective proxy for the true information bottleneck as shown in the experiments in Kirsch et al. (2020) (e.g., see their experiment on Imagenette dataset). Also, our experiments validate this approximation even in moderately high dimensions, as an example in CS-CMNIST, the dimension of the layer at which bottleneck constraints are applied is 256. Developing tighter approximations for information bottleneck in high dimensions and analyzing their impact on OOD generalization is an important future work. In recent works (Rosenfeld et al., 2021b; Kamath et al., 2021; Gulrajani and Lopez-Paz, 2021), there has been criticism of different aspects of IRM, e.g., failure of IRMv1 penalty in non-linear models, the tuning of IRMv1 penalty, etc. Since we use IRMv1 penalty in our proposed loss, these criticisms apply to our objective as well. Other approximations of invariance have been proposed in the literature (Koyama and Yamaguchi, 2020; Ahuja et al., 2020; Chang et al., 2020). Exploring their benefits together with information bottleneck is a fruitful future work. Before concluding, we want to remark that we have already discussed the closest related works. However, we also provide a detailed discussion of the broader related literature in the Appendix.

7 Conclusion

In this work, we revisited the fundamental assumptions for OOD generalization for settings when invariant features capture all the information about the label. We showed how linear classification tasks are different and need much stronger assumptions than linear regression tasks. We provide a sharp characterization of performance of ERM and IRM under different assumptions on support overlap of invariant and spurious features. We showed that support overlap of invariant features is necessary or otherwise OOD generalization is impossible. However, ERM and IRM seem to fail even in the absence of support overlap of spurious features. We prove that a form of the information bottleneck constraint along with invariance goes a long way in overcoming the failures while retaining the existing provable guarantees.

Acknowledgements

We thank Reyhane Askari Hemmat, Adam Ibrahim, Alexia Jolicoeur-Martineau, Divyat Mahajan, Ryan D'Orazio, Nicolas Loizou, Manuela Girotti, and Charles Guille-Escuret for the feedback. Kartik Ahuja would also like to thank Karthikeyan Shanmugam for discussions pertaining to the related works.

Funding disclosure

We would like to thank Samsung Electronics Co., Ldt. for funding this research. Kartik Ahuja acknowledges the support provided by IVADO postdoctoral fellowship funding program. Yoshua Bengio acknowledges the support from CIFAR and IBM. Ioannis Mitliagkas acknowledges support from an NSERC Discovery grant (RGPIN-2019-06512), a Samsung grant, Canada CIFAR AI chair and MSR collaborative research grant. Irina Rish acknowledges the support from Canada CIFAR AI Chair Program and from the Canada Excellence Research Chairs Program. We thank Compute Canada for providing computational resources.

References

Ahmed, F., Bengio, Y., van Seijen, H., and Courville, A. (2021). Systematic generalisation with group invariant predictions. In *International Conference on Learning Representations*.

Ahuja, K., Shanmugam, K., and Dhurandhar, A. (2021a). Linear regression games: Convergence guarantees to approximate out-of-distribution solutions. In *International Conference on Artificial Intelligence and Statistics*, pages 1270–1278. PMLR.

Ahuja, K., Shanmugam, K., Varshney, K., and Dhurandhar, A. (2020). Invariant risk minimization games. In *International Conference on Machine Learning*, pages 145–155. PMLR.

Ahuja, K., Wang, J., Dhurandhar, A., Shanmugam, K., and Varshney, K. R. (2021b). Empirical or invariant risk minimization? a sample complexity perspective. In *International Conference on Learning Representations*.

Albuquerque, I., Monteiro, J., Falk, T. H., and Mitliagkas, I. (2019). Adversarial target-invariant representation learning for domain generalization. *arXiv preprint arXiv:1911.00804*.

Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. (2016). Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*.

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

Arpit, D., Xiong, C., and Socher, R. (2019). Entropy penalty: Towards generalization beyond the iid assumption.

Ash, R. B. and Doléans-Dade, C. A. (2000). *Probability and Measure Theory*. Academic Press, San Diego, California.

Aubin, B., Słowik, A., Arjovsky, M., Bottou, L., and Lopez-Paz, D. (2021). Linear unit-tests for invariance discovery. *arXiv preprint arXiv:2102.10867*.

Beery, S., Van Horn, G., and Perona, P. (2018). Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision*, pages 456–473.

Ben-David, S., Blitzer, J., Crammer, K., Kulesza, A., Pereira, F., and Vaughan, J. W. (2010). A theory of learning from different domains. *Machine learning*, 79(1-2):151–175.

Ben-David, S., Blitzer, J., Crammer, K., and Pereira, F. (2007). Analysis of representations for domain adaptation. In *Advances in neural information processing systems*, pages 137–144.

Ben-David, S. and Urner, R. (2012). On the hardness of domain adaptation and the utility of unlabeled target samples. In *International Conference on Algorithmic Learning Theory*, pages 139–153. Springer.

Chang, S., Zhang, Y., Yu, M., and Jaakkola, T. S. (2020). Invariant rationalization. In *International Conference on Machine Learning, 2020*.

David, S. B., Lu, T., Luu, T., and Pál, D. (2010). Impossibility theorems for domain adaptation. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics*, pages 129–136. JMLR Workshop and Conference Proceedings.

DeGrave, A. J., Janizek, J. D., and Lee, S.-I. (2020). AI for radiographic COVID-19 detection selects shortcuts over signal. *medRxiv*.

Deng, Z., Ding, F., Dwork, C., Hong, R., Parmigiani, G., Patil, P., and Sur, P. (2020). Representation via representations: Domain generalization via adversarially learned invariant representations. *arXiv preprint arXiv:2006.11478*.

Ganin, Y., Ustinova, E., Ajakan, H., Germain, P., Larochelle, H., Laviolette, F., Marchand, M., and Lempitsky, V. (2016). Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(1):2096–2030.

Garg, V., Kalai, A. T., Ligett, K., and Wu, S. (2021). Learn to expect the unexpected: Probably approximately correct domain generalization. In *International Conference on Artificial Intelligence and Statistics*, pages 3574–3582. PMLR.

Geirhos, R., Jacobsen, J.-H., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., and Wichmann, F. A. (2020). Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673.

Greenfeld, D. and Shalit, U. (2020). Robust learning with the hilbert-schmidt independence criterion. In *International Conference on Machine Learning*, pages 3759–3768. PMLR.

Gulrajani, I. and Lopez-Paz, D. (2021). In search of lost domain generalization. In *International Conference on Learning Representations*.

Heinze-Deml, C., Peters, J., and Meinshausen, N. (2018). Invariant causal prediction for nonlinear models. *Journal of Causal Inference*, 6(2).

Huszár, F. (2019). https://www.inference.vc/invariant-risk-minimization/.

Jin, W., Barzilay, R., and Jaakkola, T. (2020). Enforcing predictive invariance across structured biomedical domains.

Kamath, P., Tangella, A., Sutherland, D. J., and Srebro, N. (2021). Does invariant risk minimization capture invariance? *arXiv preprint arXiv:2101.01134*.

Khalil, H. K. (2009). Lyapunov stability. *Control Systems, Robotics and AutomatioN–Volume XII: Nonlinear, Distributed, and Time Delay Systems-I*, page 115.

Khemakhem, I., Kingma, D., Monti, R., and Hyvarinen, A. (2020). Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217. PMLR.

Kirsch, A., Lyle, C., and Gal, Y. (2020). Unpacking information bottlenecks: Unifying information-theoretic objectives in deep learning. *arXiv preprint arXiv:2003.12537*.

Koyama, M. and Yamaguchi, S. (2020). Out-of-distribution generalization with maximal invariant predictor. *arXiv preprint arXiv:2008.01883*.

Krueger, D., Caballero, E., Jacobsen, J.-H., Zhang, A., Binas, J., Zhang, D., Priol, R. L., and Courville, A. (2020). Out-of-distribution generalization via risk extrapolation (rex). *arXiv preprint arXiv:2003.00688*.

Li, Y., Tian, X., Gong, M., Liu, Y., Liu, T., Zhang, K., and Tao, D. (2018). Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 624–639.

Lu, C., Wu, Y., Hernández-Lobato, J. M., and Schölkopf, B. (2021). Nonlinear invariant risk minimization: A causal approach. *arXiv preprint arXiv:2102.12353*.

Mahajan, D., Tople, S., and Sharma, A. (2020). Domain generalization using causal matching. *arXiv preprint arXiv:2006.07500*.

Matsuura, T. and Harada, T. (2020). Domain generalization using a mixture of multiple latent domains. In *AAAI*, pages 11749–11756.

Muandet, K., Balduzzi, D., and Schölkopf, B. (2013). Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pages 10–18.

Müller, J., Schmier, R., Ardizzone, L., Rother, C., and Köthe, U. (2020). Learning robust models using the principle of independent causal mechanisms. *arXiv preprint arXiv:2010.07167*.

Nagarajan, V., Andreassen, A., and Neyshabur, B. (2021). Understanding the failure modes of out-of-distribution generalization. In *International Conference on Learning Representations*.

Pagnoni, A., Gramatovici, S., and Liu, S. (2018). Pac learning guarantees under covariate shift. *arXiv preprint arXiv:1812.06393*.

Parascandolo, G., Neitz, A., ORVIETO, A., Gresele, L., and Schölkopf, B. (2021). Learning explanations that are hard to vary. In *International Conference on Learning Representations*.

Pearl, J. (1995). Causal diagrams for empirical research. *Biometrika*, 82(4):669–688.

Pearl, J. (2009). *Causality*. Cambridge university press.

Peters, J., Bühlmann, P., and Meinshausen, N. (2015). Causal inference using invariant prediction: identification and confidence intervals. *arXiv preprint arXiv:1501.01332*.

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

Pezeshki, M., Kaba, S.-O., Bengio, Y., Courville, A., Precup, D., and Lajoie, G. (2020). Gradient starvation: A learning proclivity in neural networks. *arXiv preprint arXiv:2011.09468*.

Piratla, V., Netrapalli, P., and Sarawagi, S. (2020). Efficient domain generalization via common-specific low-rank decomposition. In *International Conference on Machine Learning, 2020*.

Redko, I., Morvant, E., Habrard, A., Sebban, M., and Bennani, Y. (2019). *Advances in Domain Adaptation Theory*. Elsevier.

Robey, A., Pappas, G. J., and Hassani, H. (2021). Model-based domain generalization. *arXiv preprint arXiv:2102.11436*.

Rojas-Carulla, M., Schölkopf, B., Turner, R., and Peters, J. (2018). Invariant models for causal transfer learning. *The Journal of Machine Learning Research*, 19(1):1309–1342.

Rosenfeld, E., Ravikumar, P., and Risteski, A. (2021a). An online learning approach to interpolation and extrapolation in domain generalization. *arXiv preprint arXiv:2102.13128*.

Rosenfeld, E., Ravikumar, P. K., and Risteski, A. (2021b). The risks of invariant risk minimization. In *International Conference on Learning Representations*.

Sagawa, S., Koh, P. W., Hashimoto, T. B., and Liang, P. (2019). Distributionally robust neural networks. In *International Conference on Learning Representations*.

Schölkopf, B., Janzing, D., Peters, J., Sgouritsa, E., Zhang, K., and Mooij, J. (2012). On causal and anticausal learning. *arXiv preprint arXiv:1206.6471*.

Simmons, G. F. (2016). *Differential equations with applications and historical notes*. CRC Press.

Soudry, D., Hoffer, E., Nacson, M. S., Gunasekar, S., and Srebro, N. (2018). The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878.

Strouse, D. and Schwab, D. J. (2017). The deterministic information bottleneck. *Neural computation*, 29(6):1611–1630.

Teney, D., Abbasnejad, E., and Hengel, A. v. d. (2020). Unshuffling data for improved generalization. *arXiv preprint arXiv:2002.11894*.

Tishby, N., Pereira, F. C., and Bialek, W. (2000). The information bottleneck method. *arXiv preprint physics/0004057*.

Vapnik, V. (1992). Principles of risk minimization for learning theory. In *Advances in neural information processing systems*, pages 831–838.

Xie, S. M., Kumar, A., Jones, R., Khani, F., Ma, T., and Liang, P. (2021). In-n-out: Pre-training and self-training using auxiliary information for out-of-distribution robustness. In *International Conference on Learning Representations*.

Zhang, D., Ahuja, K., Xu, Y., Wang, Y., and Courville, A. C. (2021). Can subnetwork structure be the key to out-of-distribution generalization? In *ICML*.

Zhao, H., Combes, R. T. d., Zhang, K., and Gordon, G. J. (2019). On learning invariant representation for domain adaptation. *arXiv preprint arXiv:1901.09453*.

Zhao, S., Gong, M., Liu, T., Fu, H., and Tao, D. (2020). Domain generalization via entropy regularization. *Advances in Neural Information Processing Systems*, 33.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes] See Section 2-5 and the additional details such as the proofs in the supplementary material.
   (b) Did you describe the limitations of your work? [Yes] See Section 4.1 and Section 6.
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section A.1 in the Appendix in the supplementary material.
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 2-4.
   (b) Did you include complete proofs of all theoretical results? [Yes] See the Appendix in the Supplementary Material.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] See `https://github.com/ahujak/IB-IRM`
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section A.2 in the Appendix in the supplementary material.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Section A.2 in the Appendix in the supplementary material.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section A.2 in the Appendix in the supplementary material.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] We use the codes from following github repositories `https://github.com/facebookresearch/DomainBed`, `https://github.com/facebookresearch/InvariantRiskMinimization` and `https://github.com/facebookresearch/InvarianceUnitTests` and we have cited the creators in the Section A.2 in the Appendix in the supplementary material.

(b) Did you mention the license of the assets? [Yes] All the repositories mentioned above use MIT license. We have mentioned this in Section A.2 in the Appendix in the supplementary material.

(c) Did you include any new assets either in the supplemental material or as a URL? [Yes] We have included code for our experiments in the supplementary material.

(d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

(e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

A Appendix

Organization. In Section A.1, we discuss the societal impact of this work. In Section A.2, we provide further details on the experiments. In Section A.3, we provide a detailed discussion on structural equation models and the linear general position assumption used to prove Theorem 1. In Section A.4, we first cover the notations used in the proofs, followed by some technical remarks to be kept in mind for all the proofs, and then we provide the proof of the impossibility result in Theorem 2. In Section A.5, we provide the proof for sufficiency and insufficiency characterization of ERM and IRM discussed in Theorem 3. In Section A.6, we provide the proof for Theorem 4, which compares IB-IRM, IB-ERM with IRM and ERM. In Section A.7, we discuss the step-by-step derivation of the final objective in equation (7). In Section A.8, we provide the proof for Theorem 5, which compares the impact of information bottleneck penalty on the learning speed. In Section A.9, we provide an analysis of settings when both IRM and IB penalty work together in conjunction. Also, at the end of each section describing a proof, we provide remarks on various aspects, including some simple extensions that our results already cover. Although in the main manuscript we covered the relevant related works, in Section A.10, we provide a more detailed discussion on other related works.

A.1 Societal impact

When machine learning models are deployed to assist in making decisions in safety-critical applications (e.g., self-driving cars, healthcare, etc.), we want to ensure that they make decisions that can be trusted well beyond the regime of the training data that they are exposed to. The models used in current practice are prone to exploiting spurious correlations/shortcuts in arriving at decisions and are thus not always reliable. In this work, we took some steps towards building a well-founded theory and proposing methods based on the same that can eventually help us build machines that work well beyond the training data regime. At this point, we do not anticipate a negative impact specifically of this work.

A.2 Experiments details

In this section, we provide further details on the experiments. The codes to reproduce the experiments is provided at `https://github.com/ahujak/IB-IRM`. We have also added the codes to DomainBed (`https://github.com/facebookresearch/DomainBed`).

A.2.1 Datasets

We first describe the datasets (Example 1/1S, Example 2/2S, Example 3/3S) introduced in Aubin et al. (2021); these datasets are referred to as the linear unit tests. The results for linear unit tests are presented in Table 3.

Example 1/1S (PIIF). This example follows the linear regression SEM from Assumption 1. The dataset in environment $e \in E_{all}$ is sampled from the following

$$
\begin{aligned}
Z^e_{\text{inv}} &\sim N_m(0; (\sigma^e)^2); & Y^e &\sim N_m(W_{yz} Z^e_{\text{inv}}; (\sigma^e)^2);\\
Z^e_{\text{spu}} &\sim N_o(W_{zy} Y^e; 1); & Z^e &= (Z^e_{\text{inv}}; Z^e_{\text{spu}});\\
Y^e &= \frac{2}{(m+o)} 1_m^\top Y^e; & X^e &= S(Z^e);
\end{aligned}
$$

where $W_{yz} \in \mathbb{R}^{m \times m}$, $W_{zy} \in \mathbb{R}^{o \times m}$ are matrices drawn i.i.d. from the standard normal distribution, $1_m \in \mathbb{R}^m$ is a vector of ones, N_k is a k dimensional vector from the normal distribution. For the first three environments $(e_0; e_1; e_2)$, the variances are fixed as $(\sigma^{e_0})^2 = 0.1$, $(\sigma^{e_1})^2 = 1.5$, and $(\sigma^{e_2})^2 = 2.0$. When the number of environments is greater than three, then $(\sigma^{e_j})^2 \sim \text{Uniform}(10^{-2}; 10)$. The scrambling matrix S is set to identity in Example 1 and a random unitary matrix is selected to rotate the latents in Example 1S. In the above dataset, the invariant features are causal and partially informative about the label. The spurious features are anti-causally related to the label and carry extra information about the label not contained in the invariant features.

Example 2/2S (FIIF). This example follows the linear classification SEM from Assumption 2 with zero noise. The dataset generalizes the 2D cow versus camel classification task in equation (4). Let

$$\text{cow} = 1_m; \quad \text{camel} = \text{cow}; \quad \text{animal} = 10^{-2};$$
$$\text{grass} = 1_o; \quad \text{sand} = \text{grass}; \quad \text{background} = 1.$$

The dataset in environment $e \in E_{all}$ is sampled from the following distribution

$$\begin{aligned}
U^e &\quad \text{Categorical}\ p^e s^e, (1-p^e)s^e, p^e(1-s^e), (1-p^e)(1-s^e)\ ; \\
Z^e_{\text{inv}} &\quad \begin{cases} (N_m(0, 0.1) + \text{cow}) \cdot \text{animal} & \text{if } U^e \in \{1, 2\}; \\ (N_m(0, 0.1) + \text{camel}) \cdot \text{animal} & \text{if } U^e \in \{3, 4\}; \end{cases} \\
Z^e_{\text{spu}} &\quad \begin{cases} (N_o(0, 0.1) + \text{grass}) \cdot \text{background} & \text{if } U^e \in \{1, 4\}; \\ (N_o(0, 0.1) + \text{sand}) \cdot \text{background} & \text{if } U^e \in \{2, 3\}; \end{cases} \\
Z^e &\quad (Z^e_{\text{inv}}, Z^e_{\text{spu}}); \quad X^e \quad S(Z^e); \\
Y^e &\quad \mathbb{I}(1_m^T Z^e_{\text{inv}});
\end{aligned}$$

where for the first three environments the background parameters are $p^{e_0} = 0.95$, $p^{e_1} = 0.97$, $p^{e_2} = 0.99$ and the animal parameters are $s^{e_0} = 0.3$, $s^{e_1} = 0.5$, $s^{e_2} = 0.7$. When the number of environments are greater than three, then $p^{e_j} \quad \text{Uniform}(0.9, 1)$, and $s^{e_j} \quad \text{Uniform}(0.3, 0.7)$. The scrambling matrix S is set to identity in Example 2 and a random unitary matrix is selected to rotate the latents in Example 2S. In the above dataset, the invariant features are causal and carry full information about the label. The spurious features are correlated with the invariant features through a confounding selection bias U^e.

Example 3/3S (PIIF). This example is a classification problem following the SEM assumed in (Rosenfeld et al., 2021b). The example is meant to present a linear version of the spiral classification problem in (Parascandolo et al., 2021). Let $\text{inv} = 0.1 \quad 1_m$, and $\text{spu} \quad N_o(0, 1)$ for all the environments. The dataset in environment $e \in E_{all}$ is sampled from the following distribution

$$\begin{aligned}
Y^e &\quad \text{Bernoulli}\ \frac{1}{2}\ ; \\
Z^e_{\text{inv}} &\quad \begin{cases} N_m(+\ \text{inv}, 0.1) \text{ if } Y^e = 0; \\ N_m(\ \ \text{inv}, 0.1) \text{ if } Y^e = 1; \end{cases} \\
Z^e_{\text{spu}} &\quad \begin{cases} N_o(+\ \text{spu}, 0.1) \text{ if } Y^e = 0; \\ N_o(\ \ \text{spu}, 0.1) \text{ if } Y^e = 1; \end{cases} \quad ; \\
Z^e &\quad (Z^e_{\text{inv}}, Z^e_{\text{spu}}); \quad X^e \quad S(Z^e).
\end{aligned} \tag{8}$$

The scrambling matrix S is set to identity in Example 3 and a random unitary matrix is selected to rotate the latents in Example 3S. In the above dataset, the invariant features are anti-causally related to the label Y^e. The spurious features carry extra information about the label not contained in the invariant features.

AC-CMNIST dataset (PIIF). We follow the same construction as was proposed in Arjovsky et al. (2019). We set up a binary classification task– identify whether the digit is less than 5 (not including 5) or more than 5. There are three environments – two training environments containing 25,000 data points each, one test environment containing 10,000 points. Define a preliminary label $\tilde{Y} = 0$ if the digit is between 0-4 and $\tilde{Y} = 1$ if the digit is between 5-9. We add noise to this preliminary label by flipping it with a 25 percent probability to construct the final label. We flip the final labels to obtain the color id Z^e_{spu}, where the flipping probabilities are environment-dependent. The flipping probabilities are 0.2, 0.1, and 0.9, in the first, second, and third environment respectively. The third environment is the testing environment. If $Z^e_{\text{spu}} = 1$, we color the digit red, otherwise we color it to be green. In this dataset, the color (spurious feature) carries extra information about the label not contained in the uncolored image.

CS-CMNIST dataset (FIIF). We follow the same construction based on Ahuja et al. (2021b), except instead of a binary classification task, we set up a ten-class classification task, where the ten classes are

the ten digits. For each digit class, we have an associated color.[10] There are also three environments – two training environments containing 20,000 data points each, one test containing 20,000 points. In the two training environments, the p^e is set to 1:0 and 0:9, i.e., given the digit label the image is colored with the associated color with probability p^e and with a random color with probability 1 p^e. In the testing environment, the p^e is set to 0, i.e., all the images are colored completely at random. In this dataset, the color (spurious feature) does not carry any extra information about the label that is not already contained in the uncolored image.

Terra Incognita dataset (FIIF). This dataset is a subset of the Caltech Camera Traps dataset (Beery et al., 2018) as formulated in Gulrajani and Lopez-Paz (2021). We set up a ten-class classification task for 3 224 224 images - identifying between 9 different species of wild animal and no animal ({ bird, bobcat, cat, coyote, dog, empty, opossum, rabbit, raccoon, squirrel}). There are four domains - {L100, L38, L43, L46} - which represents different locations of the cameras in the American Southwest. For a given location the background never change, except for illumination difference across the time of day and vegetation changes across seasons. The data is unbalanced in the number of images per location, distribution of species per location, and distribution of species overall.

COCO dataset (FIIF). We use COCO on colours dataset described in Ahmed et al. (2021) (See the details in Appendix A.2 of Ahmed et al. (2021)). There are ten object classes and for each object class there is a majority color associated with it, i.e., an object class assumes the background color assigned to it with 0:8 probability. At test time, the object backgrounds are colored randomly with colors different from the ones seen in training.

A.2.2 Training and evaluation procedure

Example 1/1S, 2/2S, 3/3S. We follow the same protocol as was prescribed in Aubin et al. (2021) for the model selection, hyperparameter selection, training, and evaluation. For all three examples, the models used are linear. The training loss is the square error for the regression setting (Example 1/1S), and binary cross-entropy for the classification setting (Example 2/2S, 3/3S). For the two new approaches, IB-IRM, and IB-ERM, there is a new hyperparameter associated with the Var( ) term in the final objective in equation (7). We use random hyperparameter search and use 20 hyperparameter queries and average over 50 data seeds; these numbers are the same as what was used in Aubin et al. (2021). We sample the from 1 $10^{\text{Uniform}(-2;0)}$ following the practice in unit test experiments (Aubin et al., 2021). Note that the hyperparameters are trained using training environment distribution data, which is called the train-domain validation set evaluation procedure in Gulrajani and Lopez-Paz (2021). For the evaluation of performance on Example 1/1s, we reported mean square errors and standard deviations. For the evaluation of performance on Example 2/2S, Example 3/3s, we reported classification errors and standard deviations.

AC-CMNIST dataset. We use the default MLP architecture from `https://github.com/facebookresearch/InvariantRiskMinimization`. There are two fully connected layers each with output size 256, ReLU activation, and ℓ_2-regularizer coefficient of 1e 3. These layers are followed by the output layer of size two. We use Adam optimizer for training with a learning rate set to 1e 3. We optimize the cross-entropy loss function. We set the batch size to 256. The total number of steps is set to 500. We use grid search to search the following hyperparameters, for IRMv1 penalty, and for the IB penalty. For IRM, we need to select the IRMv1 penalty , we set a grid of 25 values uniformly spaced in the interval [1e 1; 1:8e4]. For IB-ERM, we need to select the IB penalty , we set a grid of 25 values uniformly spaced in the interval [1e 1; 1:8e4]. For IB-IRM, we need to select both and , we set a 5 5 uniform grid that searches over [1e 1; 1:8e4] [1e 1; 1:8e4]. Thus for IB-IRM, IB-ERM, and IRM, we search over 25 hyperparameter values. There are two procedures we tried to tune the hyperparameters – a) train-domain validation set tuning procedure (Gulrajani and Lopez-Paz, 2021) which takes samples from the same distribution as train domain and does limited model queries (we set 25 queries), b) oracle test-domain validation set hyperparameter tuning procedure (Gulrajani and Lopez-Paz, 2021), which takes samples from the same distribution as test domain and does limited model queries (we set 25 queries). In Arjovsky et al. (2019), the authors had used oracle test-domain validation set-based tuning, which is not ideal and is a limitation of all current approaches on AC-CMNIST. We used the same procedure in Table 4 (5 percent of the total data 50000 follows the test environment distribution). In Section A.2.3, we show the results for

[10] The list of the RGB values for the ten colors are: [0, 100, 0], [188, 143, 143], [255, 0, 0], [255, 215, 0], [0, 255, 0], [65, 105, 225], [0, 225, 225], [0, 0, 255], [255, 20, 147], [160, 160, 160].

all the methods when we use train-domain validation set tuning. For the evaluation, we reported the accuracy and standard deviations (averaged over thirty trials).

CS-CMNIST dataset. We use a ConvNet architecture with three convolutional layers with feature map dimensions of 64,128 and 256. Each convoluional layer is followed by a ReLU activation and batch normalization layer. The final output layer is a linear layer with output dimension equal to the number of classes. We use SGD optimizer for training with a learning rate set to 1e 1 and decay every 600 steps. We optimize the cross-entropy loss function without weight decay. We set the batch size to 128. The total number of steps is set to 2000. We use grid search to search the following hyperparameters, for IRMv1 penalty, and for the IB penalty. For IRM, we need to select the IRMv1 penalty , we set a grid of 25 values uniformly spaced in the interval [1e 1; 1:8e4]. For IB-ERM, we need to select the IB penalty , we set a grid of 25 values uniformly spaced in the interval [1e 1; 1:8e4]. For IB-IRM, we need to select both and , we set a 5 5 uniform grid that searches over [1e 1; 1:8e4] [1e 1; 1:8e4]. Thus for IB-IRM, IB-ERM, and IRM, we search over 25 hyperparameter values. In the paragraph above, we described that for AC-CMNIST all the procedures only work when using the oracle test-domain validation procedure. In the results of the CS-CMNIST experiment in the main manuscript, we showed results for the train domain validation procedure and found that IB-IRM and IB-ERM yield better performance. For completeness, we also carried oracle test-domain validation procedure-based hyperparameter tuning for CS-CMNIST and the results are discussed in Section A.2.3. For the evaluation, we reported accuracy and standard deviations (averaged over five trials). In both CMNIST datasets, we had experimented with placing the IB penalty at the output layer (logits) and the penultimate layer (layer just before the logits), and found that it is much more effective to place the IB penalty on the penultimate layer. Thus in both the CMNIST datasets, the results presented use IB penalty on the penultimate layer.

Terra Incognita dataset. We use the pretrained ResNet-50 model as a featurizer that outputs feature maps of size 2048 for a given image on top of which we add a 1 layer MLP which makes the classification (2048 ! 9). We use a random hyper parameter sweep over 20 random hyperparameter configurations on which we look at the train-domain validation set to perform model selection, as described in Gulrajani and Lopez-Paz (2021). The distribution of the hyper parameters are shown in Table 5. Results shown in Table 4 are for the environment L100 as test environment, the reported accuracies are averaged over 3 random trial seed. For both the information bottleneck penalized algorithms (IB-ERM and IB-IRM), we apply the penalty on the feature map given by the featurizer, conditional on the environment.

Table 5: Hyperparameters distributions for random search given included penalty of the algorithm.

Penalty	**Parameter**	**Random distribution**
All	dropout	RandomChoice([0; 0:1; 0:5])
	learning rate	$10^{\text{Uniform}(-5;-3:5)}$
	batch size	$2^{\text{Uniform}(3;5:5)}$
	weight decay	$10^{\text{Uniform}(-6;-2)}$
IRMv1	penalty weight	$10^{\text{Uniform}(-1;5)}$
	annealing steps	$10^{\text{Uniform}(0;4)}$
IB	penalty weight	$10^{\text{Uniform}(-1;5)}$
	annealing steps	$10^{\text{Uniform}(0;4)}$

COCO dataset. Other than the IB penalty, we use the exact same hyperparameters (default values) and setup as describe in Appendix B.2 of Ahmed et al. (2021) paper and the codebase that Ahmed et al. (2021) paper provides. For all experiments that involve an IB loss term component, IB penalty weighting of 1.0 is used and IB penalty weighting is linearly ramped up to 1.0 from epoch 1 to 200. For all experiments that involve an IRM loss term component, IRM penalty weighting of 1.0 is used, and IRM penalty weighting is linearly ramped up to 1.0 from epoch 1 to 200. Batch size of 64 is used for all experiments. We do not tune the hyperparameters in this experiment. Mean and standard deviation of classification accuracy are obtained via 4 seeds for each method.

A.2.3 Supplementary experiments

AC-CMNIST. In the AC-CMNIST dataset, for completeness, we report the accuracy of the Oracle model, where the Oracle model at train time is fed images where the background colors do not have any correlation with the label. Oracle model achieved a test accuracy 70:39 0:47 percent. In Table 5, we provide the supplementary experiments for AC-CMNIST carried out with train-domain validation set tuning procedure (Gulrajani and Lopez-Paz, 2021). It can be seen that none of the methods work in this case. In Table 6, we provide the supplementary experiments for AC-CMNIST carried out with test-domain validation set tuning procedure (Gulrajani and Lopez-Paz, 2021). In this case, both IB-IRM and IRM perform well.

Method	5%		10%		15%		20%	
ERM	17:17	0:62	18:06	1:72	18:74	1:23	19:11	1:18
IB-ERM	17:69	0:54	17:80	1:81	16:27	1:20	18:18	1:46
IRM	16:48	2:50	17:85	1:67	17:32	2:12	18:09	2:78
IB-IRM	18:37	1:44	17:83	0:65	18:54	1:42	19:24	1:49

Table 6: AC-CMNIST. Comparisons of the methods using the train-domain validation set tuning procedure (Gulrajani and Lopez-Paz, 2021). The percentages in the columns indicate what fraction of the total data (50000 points) is used for validation.

Method	5%		10%		15%		20%	
ERM	16:84	0:82	17:01	0:83	16:79	0:89	16:27	0:93
IB-ERM	50:24	0:47	50:25	0:46	50:52	0:45	50:34	0:56
IRM	66:98	1:65	67:57	1:39	67:01	1:86	67:29	1:62
IB-IRM	67:67	1:78	68:22	1:62	67:56	1:71	67:24	1:36

Table 7: CS-CMNIST. Comparisons of the methods using the oracle test-domain validation set tuning procedure (Gulrajani and Lopez-Paz, 2021). The percentages in the columns indicate what fraction of the total data (50000 points) is used for validation.

AC-CMNIST. In the CS-CMNIST dataset, for completeness, we report the accuracy of the Oracle model, which achieved a test accuracy of 99:03 0:08 percent. In Table 7, we provide the supplementary experiments for CS-CMNIST carried out with train-domain validation set tuning procedure (Gulrajani and Lopez-Paz, 2021). In Table 8, we provide the supplementary experiments for CS-CMNIST carried out with test-domain validation set tuning procedure (Gulrajani and Lopez-Paz, 2021). In both cases, both IB-IRM and IB-ERM RM perform well. Unlike AC-CMNIST, in the CS-CMNIST dataset both the validation procedures lead to a similar performance.

Method	5%		10%		15%		20%	
ERM	60:27	1:21	61:02	0:59	60:35	1:01	58:59	1:67
IB-ERM	71:80	0:69	71:51	1:01	71:27	1:04	70:68	1:02
IRM	61:49	1:45	61:74	1:28	60:01	0:59	59:96	0:96
IB-IRM	71:79	0:70	71:57	1:01	71:37	0:62	70:65	0:90

Table 8: CS-CMNIST. Comparisons of the methods using the train-domain validation set tuning procedure (Gulrajani and Lopez-Paz, 2021). The percentages in the columns indicate what fraction of the total data (50000 points) is used for validation.

Method	5%		10%		15%		20%	
ERM	61:27	1:40	61:02	1:59	60:35	1:01	58:59	1:67
IB-ERM	71:65	0:76	71:68	1:23	71:27	0:89	70:07	1:18
IRM	62:00	1:60	62:01	1:33	60:26	0:51	59:96	0:96
IB-IRM	71:90	0:78	71:07	0:95	71:18	0:80	70:75	1:00

Table 9: CS-CMNIST. Comparisons of the methods using the oracle test-domain validation set tuning procedure (Gulrajani and Lopez-Paz, 2021). The percentages in the columns indicate what fraction of the total data (50000 points) is used for validation

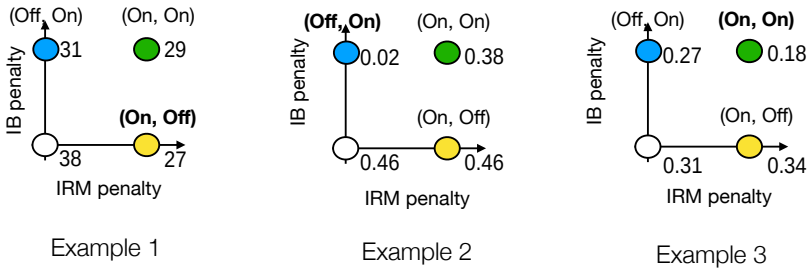


Figure 4: Illustrating the impact of the IB and IRM penalty on linear unit tests (Aubin et al., 2021)

Ablation to understand the role of invariance penalty and information bottleneck. In the main body, we compared IB-IRM, IB-ERM, IRM, and ERM with the penalty of the respective methods tuned using the validation procedures from Gulrajani and Lopez-Paz (2021). In this section, we carry out an ablation analysis on linear unit tests (Aubin et al., 2021) to understand the role of the different penalties. In Figure 4, for each example we consider the setting with six environments and show four points on a square with corresponding performance values. The bottom corner corresponds to ERM when both penalties are turned off, top corner is when both penalties are turned on, and the other two corners are when one of the penalties are on. In Example 1, which corresponds to PIIF setting, we find that IRM penalty alone helps the most. In Example 2, which corresponds to FIIF setting, we find that IB penalty helps the most. In Example 3, which again corresponds to PIIF, we find that both penalties help.

A.2.4 Compute description

Our computing resource is one Tesla V100-SXM2-16GB with 18 CPU cores.

A.2.5 Assets used and the license details

In this work, we mainly relied on the following github repositories – Domainbed[11], IRM [12], linear unit tests[13]. All the repositories mentioned above use the MIT license. We used the standard MNIST dataset [14] to generate the colored MNIST datasets. Other datasets we used are synthetic.

[11] `https://github.com/facebookresearch/DomainBed` based on Gulrajani and Lopez-Paz (2021)

[12] `https://github.com/facebookresearch/InvariantRiskMinimization` based on Arjovsky et al. (2019)

[13] `https://github.com/facebookresearch/InvarianceUnitTests` based on Aubin et al. (2021)

[14] `http://yann.lecun.com/exdb/mnist/`

A.3 Background on structural equation models

For completeness, we provide a more detailed background on structural equation models (SEMs), which is borrowed from Arjovsky et al. (2019).

A.3.1 Structural equation models and assumptions on E_{all}

Definition 1. *A structural equation model $C = (S; N)$ that describes the random vector $X = (X_1, \ldots, X_d)$ is given as follows*

$$S_i : X_i \quad f_i(\mathrm{Pa}(X_i); N_i); \tag{9}$$

where $\mathrm{Pa}(X_i)$ are the parents of X_i, N_i is independent noise, and $N = (N_1, \ldots, N_d)$ is the noise vector. X_j is said to cause X_i if $X_j \in \mathrm{Pa}(X_i)$. We draw the causal graph by placing one node for each X_i and drawing a directed edge from each parent to the child. The causal graphs are assumed to be acyclic.

Definition 2. *An intervention e on C is the process of replacing one or several of its structural equations to obtain a new intervened SEM $C^e = (S^e; N^e)$, with structural equations given as*

$$S_i^e : X_i^e \quad f_i^e(\mathrm{Pa}(X_i^e); N_i^e); \tag{10}$$

where the variable X_i^e is said to be intervened if $S_i \neq S_i^e$ or $N_i \neq N_i^e$

The above family of interventions are used to model the environments.

Definition 3. *Consider a SEM C that describes the random vector (X, Y), where $X = (X_1, \ldots, X_d)$, and the learning goal is to predict Y from X. The set of all environments obtained using interventions $E_{all}(C)$ indexes all the interventional distributions P^e, where $(X^e, Y^e) \sim P^e$. An intervention e is valid if the following conditions are met: i) the causal graph remains acyclic, ii) $\mathbb{E}[Y^e | \mathrm{Pa}(Y)] = \mathbb{E}[Y | \mathrm{Pa}(Y)]$, i.e. expectation conditional on parents is invariant, and the variance $\mathrm{Var}[Y^e | \mathrm{Pa}(Y)]$ remains within a finite range.*

Following the above definitions it is possible to show that a predictor that relies on causal parents only $v : \mathbb{R}^d \to Y$ and is given as $v(x) = \mathbb{E}[f_Y(\mathrm{Pa}(Y); N_Y)]$ solves the OOD generalization problem in equation (1) over the environments $E_{all}(C)$ that form valid interventions as stated in Definition 3. Next, we provide an example to show why v is OOD optimal.

Example to illustrate why predictors that rely on causes are robust. We reuse the toy example from Arjovsky et al. (2019) to explain why models that rely on causes are more robust to valid interventions E_{all} discussed in the previous section.

$$\begin{aligned} Y^e &\quad X_{\text{inv}}^e + \epsilon^e \\ X_{\text{spu}}^e &\quad Y^e + \epsilon^e \end{aligned} \tag{11}$$

where $X_{\text{inv}}^e \in N(0; (\epsilon^e)^2)$ is the cause of Y^e, $N^e \in N(0; (\epsilon^e)^2)$ is noise, X_{spu}^e is the effect of Y^e and $\epsilon^e \in N(0; 1)$ is also noise. Suppose there are two training environments $E_{tr} = \{e_1, e_2\}$, in the first $(\epsilon^{e_1})^2 = 1$ and in the second $(\epsilon^{e_2})^2 = 2$. The three possible models $w_{\text{inv}} X_{\text{inv}}^e + w_{\text{spu}} X_{\text{spu}}^e$ we could build are as follows: a) regress only on X_{inv}^e, then in the optimal model $w_{\text{inv}} = 1; w_{\text{spu}} = 0$, b) regress only on X_{spu}^e and get $w_{\text{inv}} = 0; w_{\text{spu}} = \frac{2}{(\epsilon^e)^2 + \frac{1}{2}}$, c) regress on $(X_{\text{inv}}^e, X_{\text{spu}}^e)$ to get $w_{\text{inv}} = \frac{1}{(\epsilon^e)^2 + 1}$ and $w_{\text{inv}} = \frac{(\epsilon^e)^2}{(\epsilon^e)^2 + 1}$. Observe that the predictor that focuses on the cause only does not depend on ϵ^2 and is thus invariant to distribution shifts induced by change in $(\epsilon^e)^2$, which is not the case with the other models. For environment in $E_{all} \cap E_{tr}$ we can change the distribution of X_{inv}^e and X_{spu}^e arbitrarily. Consider an environment $e \in E_{all}$ where X_{spu}^e is set to a very large constant c, the square error of the model that relies on spurious features grows with the magnitude of c but the error of the model that relies on X_{inv}^e does not change. Another remark we would like to make here is that in the main manuscript, we defined the notions of invariant feature map *, and spurious feature map *. Observe that in this example $^*(X^e) = X_{\text{inv}}^e$, and $^*(X^e) = X_{\text{spu}}^e$.

A.3.2 Remark on the linear general position assumption and its implications on support overlap

In Theorem 1 that we informally stated from Arjovsky et al. (2019), there is one more technical condition on that we explain below. We also explain how this assumption does not restrict the support of the latents Z^e from changing arbitrarily.

Assumption 8. ***Linear general position.*** *A set of training environments $\mathcal{E}_{tr}$ lie in a linear general position of degree r if $|\mathcal{E}_{tr}| > d - r + \frac{d}{r}$ for some $r \geq 1$ and for all non-zero $x \in \mathbb{R}^d$*

$$\dim \left(\operatorname{span} \left\{ \mathbb{E}_{X^e}[X^e X^{e\top}] x - \mathbb{E}_{X^e, Y^e}[X^e Y^e] \right\}_{e \in \mathcal{E}_{tr}} \right) > d - r. \tag{12}$$

The above assumption merely requires non-co-linearity of the training environments only. The set of matrices $\mathbb{E}_{X^e}[X^e X^{e\top}]$ not satisfying this assumption have a zero measure (Theorem 10 Arjovsky et al. (2019)). Consider the case when S is identity and observe that the above assumption translates to only a restriction on co-linearity of $\mathbb{E}_{Z^e}[Z^e Z^{e\top}]$, where $Z^e = (Z^e_{\text{inv}}, Z^e_{\text{spu}})$. Assume that $\mathbb{E}_{Z^e}[Z^e Z^{e\top}]$ is positive definite. We explain how this Assumption 8 does not constraint the support of the latent random variables Z^e. From the set of matrices $\mathbb{E}_{Z^e}[Z^e Z^{e\top}]$ and $\mathbb{E}_{Z^e}[Z^e Y^e]$ that satisfy the Assumption 8, we can construct another set of matrices with norm one that satisfy the above Assumption 8. Define a random variable $\bar{Z}^e = \frac{Z^e}{c}$ and the matrices corresponding to it also satisfy the Assumption 8, where $c = \sqrt{\|\mathbb{E}_{Z^e}[Z^e Z^{e\top}]\|}$.

For all non-zero $z \in \mathbb{R}$,

$$\begin{aligned} &\dim \left(\operatorname{span} \left\{ \mathbb{E}_{Z^e}[Z^e Z^{e\top}] z - \mathbb{E}_{Z^e, Y^e}[Z^e Y^e] \right\}_{e \in \mathcal{E}_{tr}} \right) > d - r \implies \\ &\dim \left(\operatorname{span} \left\{ \mathbb{E}_{Z^e}[Z^e Z^{e\top}] \bar{z} - \mathbb{E}_{Z^e, Y^e}[Z^e Y^e] \right\}_{e \in \mathcal{E}_{tr}} \right) > d - r; \end{aligned} \tag{13}$$

where $\bar{z} = zc$. Define $\Gamma^e = \mathbb{E}[Z^e Z^{e\top}]$ ($\tilde{\Gamma}^e = \mathbb{E}[\bar{Z}^e \bar{Z}^{e\top}]$) and $\gamma^e = \mathbb{E}[Z^e Y^e]$ ($\tilde{\gamma}^e = \mathbb{E}[\bar{Z}^e Y^e]$). Observe that $\|\tilde{\Gamma}^e\| = 1$. So far we established that if there exist a set of matrices $\{\Gamma^e, \gamma^e\}_{e \in \mathcal{E}_{tr}}$ satisfying the linear general position assumption (Assumption 8), then it also implies that there exist a set of matrices $\{\tilde{\Gamma}^e, \tilde{\gamma}^e\}_{e \in \mathcal{E}_{tr}}$, where $\|\tilde{\Gamma}^e\| = 1$, that satisfy the linear general position assumption (Assumption 8). Next, we will show that the set of matrices $\{\tilde{\Gamma}^e\}_{e \in \mathcal{E}_{tr}}$, $\{\tilde{\Gamma}^e, \tilde{\gamma}^e\}_{e \in \mathcal{E}_{tr}}$ can be constructed from random variables with bounded support. We will show that $\tilde{\Gamma}^e$ can be constructed by transforming a uniform random vector. Define a uniform random vector K^e, where each component $K^e_i \sim \text{Uniform}[-\sqrt{3}, \sqrt{3}]$. Define $Z^e = BK^e$. Observe that

$$\mathbb{E}[Z^e Z^{e,\top}] = BB^t. \tag{14}$$

Since every positive definite matrix can be decomposed as BB^t, we can use matrix B to construct the required $\tilde{\Gamma}^e$. Since $\|\tilde{\Gamma}^e\| = 1$, we get $\|BB^t\| = 1 \implies \|B\| = 1$. Also, $\|Z^e\| \leq \|B\| \|K^e\| = \|K^e\|$. Having fixed the matrix B above, we use it to set the correlation $\mathbb{E}[K^e Y^e]$

$$B\mathbb{E}[K^e Y^e] = \tilde{\gamma}^e \implies \mathbb{E}[K^e Y^e] = B^{-1}\tilde{\gamma}^e \tag{15}$$

Thus we can conclude without loss of generality that from any set of matrices $\{\Gamma^e, \gamma^e\}_{e \in \mathcal{E}_{tr}}$ satisfying the linear general position assumption, we can construct random variables with bounded support that satisfy the linear general position assumption. By solving IRM (equation (3)) over such training environments with bounded support, we can still recover the ideal invariant predictor that solves the OOD generalization problem in equation (1) (i.e., $\forall e \in \mathcal{E}_{all}$ for which risk $> \gamma^2_{\text{sup}}$). The above conditions show that we can have the data in $\mathcal{E}_{tr}$ come from a region with bounded support, and the environments in $\mathcal{E}_{all} \cap \mathcal{E}_{tr}$ are not required to satisfy support overlap with data from $\mathcal{E}_{tr}$, which is in stark contrast to the linear classification results that we showed.

A.4 Notations and proof of Theorem 2 (impossibility of guaranteed OOD generalization for linear classification)

Notations for the proofs. We describe the common notations used in the proofs that follow. We also remind the reader of the notation from the main manuscript for convenience. is used to denote the composition of functions, is used for matrix multiplication. P^e denotes the probability distribution over the input feature values X^e, and the labels Y^e in environment e. Z^e describes the latent variables decomposed into $(Z^e_{\text{inv}}; Z^e_{\text{spu}})$. S is the matrix relating X^e and Z^e and $X^e = S(Z^e)$. w denotes a linear classifier, denotes the representation map that transforms input data into a representation, which is then fed to the classifier. I is the indicator function, which takes a value 1 when the input is greater than or equal to zero, and 0 otherwise. sgn is the sign function, which takes a value 1 when the input is greater than or equal to zero, and 1 otherwise. In all the results, except for Theorem 5, we use ` as 0-1 loss for classification, and square loss for regression. For a discrete random variable $X \in \mathsf{R}^d$, the support is defined as $X = \{x \in \mathsf{R}^d \mid \mathsf{P}_X(x) > 0\}$, where $\mathsf{P}_X(x)$ is the probability of $X = x$. For a continuous random variable $X \in \mathsf{R}^d$, the support is defined as $X = \{x \in \mathsf{R}^d \mid d\mathsf{P}_X(x) > 0\}$, where $d\mathsf{P}_X(x)$ is the Radon-Nikodym derivative of P_X w.r.t the Lebesgue measure over the completion of the Borel sets in R^d (Ash and Doléans-Dade, 2000). Z^e, Z^e_{inv}, Z^e_{spu}, and X^e are the support of Z^e, Z^e_{inv}, Z^e_{spu}, and X^e respectively in environment e.

Remark on Assumption 2. In all the proofs that follow, we assume that the dimension of invariant feature m is greater than or equal to 2. Also, all the components w^*_{inv} are non-zero without loss of generality (if some component was zero, then such a latent can be a part of Z^e_{spu}. $X = \mathsf{R}^d$ and $Y = \{0; 1\}$ for classification and $Y = \mathsf{R}$ for regression. Before we can prove Theorem 2, we need to prove intermediate lemmas needed as preliminary results for it.

Define

$$W_{\text{inv}} = \{(w_{\text{inv}}; 0) \in \mathsf{R}^{m+o} \mid \|w_{\text{inv}}\| = 1;\ \forall z_{\text{inv}} \in \cup_{e\in\mathcal{E}_{tr}} Z^e_{\text{inv}};\ \mathsf{I}(w^*_{\text{inv}} z_{\text{inv}} > 0) = \mathsf{I}(w_{\text{inv}} z_{\text{inv}} > 0)\} \tag{16}$$

This set W_{inv} defines a family of hyperplanes equivalent to the labelling hyperplane w^*_{inv} on the training environments. Define a classifier $g^* : X \to Y$ as

$$g^* = \mathsf{I}(\langle w^*_{\text{inv}}; 0\rangle \circ S^{-1}) \tag{17}$$

The classifier g^* takes X^e as input and outputs $\mathsf{I}(w^*_{\text{inv}} \cdot Z^e_{\text{inv}})$.

Lemma 1. *If we consider the set of all the environments that follow Assumption 2, then the classifier based on the labelling hyperplane g^* solves equation* (1) *and achieves a risk of q in each environment.*

Proof of Lemma 1. Observe that g^* is the classifier one would get by solving for the Bayes optimal classifier on each environment. The justification goes as follows. If $w^*_{\text{inv}} \cdot Z^e_{\text{inv}} > 0$, then $\mathsf{P}(Y^e = 0 \mid X^e) < \mathsf{P}(Y^e = 1 \mid X^e)$ (since $q < \frac{1}{2}$), which implies the prediction is 1. If $w^*_{\text{inv}} \cdot Z^e_{\text{inv}} < 0$, then $\mathsf{P}(Y^e = 1 \mid X^e) < \mathsf{P}(Y^e = 0 \mid X^e)$, which implies the prediction is 0. We show that g^* achieves an error of q in each environment.

$$\begin{aligned} R^e(g^*) &= \mathsf{E}_{Y^e}[\mathsf{I}(w^*_{\text{inv}} \cdot Z^e_{\text{inv}})] \\ &= \mathsf{E}[\mathsf{I}(w^*_{\text{inv}} \cdot Z^e_{\text{inv}}) \neq N^e] \neq \mathsf{I}(w^*_{\text{inv}} \cdot Z^e_{\text{inv}})] = q. \end{aligned} \tag{18}$$

Define F to be the set of all the maps $\mathsf{R}^d \to Y$. From the equation (18) we get,

$$\begin{aligned} &\forall e \in \mathcal{E}_{all}, \forall f \in F,\ R^e(f) \geq q, \\ \implies &\forall f \in F,\ \max_{e\in\mathcal{E}_{all}} R^e(f) \geq q, \\ \implies &\min_{f\in\mathcal{F}} \max_{e\in\mathcal{E}_{all}} R^e(f) \geq q. \end{aligned} \tag{19}$$

g^* achieves the lower bound above as it achieves an error of q in each environment. This completes the proof. □

We relax the Assumption 2 to the case where we allow for spurious features to carry extra information about the label.

Assumption 9. ***Linear classification structural equation model. (PIIF)*** *In each* $e \in E_{all}$,

$$\begin{aligned} Y^e &\leftarrow \mathbb{I}\{w^*_{\text{inv}} \cdot Z^e_{\text{inv}} \geq N^e\}; \qquad N^e \sim \text{Bernoulli}(q); q < \frac{1}{2}; \qquad N^e \perp Z^e_{\text{inv}}; \\ X^e &\leftarrow S(Z^e_{\text{inv}}, Z^e_{\text{spu}}); \end{aligned} \tag{20}$$

Observe that the SEM above in Assumption 9 is analogous the the SEM in Assumption 1. Also, observe that in the above SEM $\forall e$ such that $N^e \not\perp Z^e_{\text{spu}}$, which makes the invariant features partially informative about the label. We show that the Lemma 1 extends to the above SEMs (Assumption 9) as well.

Lemma 2. *If we consider the set of all the environments that follow Assumption 9, then* g^* *solves equation* (1) *and achieves a risk of* q *in each environment.*

Proof of Lemma 2. Consider the environment $e' \in E_{all}$, where $N^{e'} \sim (Z^{e'}_{\text{inv}}, Z^{e'}_{\text{spu}})$. Observe that in this environment g^* is a Bayes optimal classifier and achieves a risk value of q.

$$\begin{aligned} \forall f \in \mathcal{F}, R^{e'}(f) \geq q &\implies \forall f \in \mathcal{F}, \max_{e \in \mathcal{E}_{all}} R^e(f) \geq q, \\ &\implies \min_{f \in \mathcal{F}} \max_{e \in \mathcal{E}_{all}} R^e(f) \geq q \end{aligned} \tag{21}$$

g^* achieves the lower bound above as it achieves an error of q in each environment. This completes the proof. □

Lemma 3. *If Assumption 2, 3, and 7 hold, and* $m \geq 2$*, then the set* W_{inv} *(eq.* (16)*) consists of infinitely many hyperplanes that are not aligned with* w^*_{inv}.

Proof of Lemma 3. For each $z_{\text{inv}} \in \bigcup_{e \in \mathcal{E}_{tr}} Z^e_{\text{inv}}$ define $y^* = \text{sgn}(w^*_{\text{inv}} \cdot z_{\text{inv}})$.

From the definition of Inv-Margin in Assumption 7, it follows that $\exists c > 0$ such that $\forall z_{\text{inv}} \in \bigcup_{e \in \mathcal{E}_{tr}} Z^e_{\text{inv}}$

$$y^* \cdot w^*_{\text{inv}} \cdot z_{\text{inv}} \geq c. \tag{22}$$

Next, we choose a $\in \mathbb{R}^m$ that is not in the same direction as w^*_{inv}, i.e., $\nexists a \in \mathbb{R}$ such that $= a w^*_{\text{inv}}$ (such a direction always exists since $m \geq 2$). Define the margin of $w^*_{\text{inv}} +$ w.r.t labels y^* from w^*_{inv}

$$y^* \cdot w^*_{\text{inv}} \cdot z_{\text{inv}} + \cdot z_{\text{inv}}. \tag{23}$$

Using Cauchy-Schwarz inequality we get

$$|y^*(\cdot z_{\text{inv}})| = | \cdot z_{\text{inv}}| \leq \| \| \cdot \|z_{\text{inv}}\|. \tag{24}$$

Since the support of the invariant features in training set $\bigcup_{e \in \mathcal{E}_{tr}} Z^e_{\text{inv}}$ is bounded, we set the magnitude of sufficiently small to control $y^* \cdot z_{\text{inv}}$. Since $\bigcup_{e \in \mathcal{E}_{tr}} Z^e_{\text{inv}}$, is bounded $\exists z^{\text{sup}} > 0$ such that $\forall z_{\text{inv}} \in \bigcup_{e \in \mathcal{E}_{tr}} Z^e_{\text{inv}}, \|z_{\text{inv}}\| < z_{\text{sup}}$. If $\| \| \leq \frac{c}{2z^{\text{sup}}}$, then from equation (24), we get that for each $z_{\text{inv}} \in \bigcup_{e \in \mathcal{E}_{tr}} Z^e_{\text{inv}}, |y \cdot z_{\text{inv}}| \leq \frac{c}{2}$. Using this we get for each $z_{\text{inv}} \in \bigcup_{e \in \mathcal{E}_{tr}} Z^e_{\text{inv}}$

$$y^* \cdot w^*_{\text{inv}} + \cdot z_{\text{inv}} = y^* \cdot w^*_{\text{inv}} \cdot z_{\text{inv}} + y^* \cdot z_{\text{inv}} \geq y^* w_{\text{inv}} \cdot z_{\text{inv}} - |y^* \cdot z_{\text{inv}}| \geq \frac{c}{2}. \tag{25}$$

From equation (22) and (25), we have that

$$\text{sgn}(w^*_{\text{inv}} +) \cdot z_{\text{inv}} = \text{sgn}(w^*_{\text{inv}} \cdot z_{\text{inv}}) \implies \mathbb{I}\{(w^*_{\text{inv}} +) \cdot z_{\text{inv}} = \mathbb{I}\{w^*_{\text{inv}} \cdot z_{\text{inv}}\}.$$

The same condition would also hold if we normalized the classifier. As a result,

$$\frac{1}{\|w^*_{\text{inv}} + \|}(w^*_{\text{inv}} +); 0 \in W_{\text{inv}}.$$

Also, observe that we can construct infinite such vectors that belong to W_{inv}. A simple way to check this this is consider $' =$, where $\in (0, 1)$. The same condition in equation (25) also holds with replaced with $'$. We define this set as follows

$$W_{\text{inv}}( ) = \left\{ \frac{1}{\|w^*_{\text{inv}} + \|}(w^*_{\text{inv}} +); 0 \in \mathbb{R}^{m+o} \mid \in [0, 1] \right\}; \tag{26}$$

and from the reasoning presented above it follows that $W_{\text{inv}}(\ ) \quad W_{\text{inv}}$. This completes the proof.

□

We restate Theorem 2 for convenience.

Theorem 6. ***Impossibility of guaranteed OOD generalization for linear classification.*** *Suppose each $e \in E_{all}$ follows Assumption 2. If for all the training environments E_{tr}, the latent invariant features are bounded and strictly separable, i.e., Assumption 3 and 7 hold, then every deterministic algorithm fails to solve the OOD generalization (eq.* (1)*), i.e., for the output of every algorithm $\exists\, e \in E_{all}$ in which the error exceeds the minimum required value q (noise level).*

Proof of Theorem 6. Consider any algorithm, it takes the data from all the training environments as inputs and outputs a classifier. We write the algorithm as a map $F : [_{i=1}^{\infty} X \quad Y \overset{i}{:::} jE_{tr}j \text{ times } [_{i=1}^{\infty} X \quad Y \overset{i}{!} Y^{\mathcal{X}}$, where F takes as input data from each of the training environments and outputs a classifier, which takes as input a data point from X and outputs the label in Y. For datasets $fD^e g_{e\in\mathcal{E}_{tr}}$ from the different training environments the output of the learner is $F \ fD^e g_{e\in\mathcal{E}_{tr}})$. For simplicity of notation, let us denote $F \ fD^e g_{e\in\mathcal{E}_{tr}})$ as f. We first show that if $f \not\equiv g^*$, where g^* is defined in equation (17), then the learner cannot be OOD optimal. Take the point x where the $f \not\equiv g^*$. Let $z = S^{-1}(x)$. Define a test environment where $Z^e = z$ occurs with probability 1. In such an environment, the error achieved by f would be $1 \quad q$ ($\mathsf{E}[f \quad g^* \quad N^e] = \mathsf{E}[1 \quad N^e] = 1 \quad q$). As a result, f cannot solve equation (1). This observation combined with Lemma 1 leads us to the conclusion that $f = g^*$ is necessary and sufficient to solve equation (1) when E_{all} follow Assumption 2.

We define a family of classifiers using W_{inv} (from eq. (16)) as follows

$$W_{\text{inv}}^{\dagger} = \left\{ I \quad (w; 0) \quad S^{-1} \quad (w; 0) \in W_{\text{inv}} \right\}. \tag{27}$$

Next, we would like to show that the set $W_{\text{inv}}^{\dagger}$ consists of infinitely many distinct functions.

Choose any w_{inv}^{0} such that $(w_{\text{inv}}^{0}; 0) \in W_{\text{inv}}$ and $w_{\text{inv}}^{0} \not\equiv w_{\text{inv}}^{*}$. Define $g^{0} = I \quad (w_{\text{inv}}^{0}; 0) \quad S^{-1}$. We will next show that $g^* \not\equiv g^{0}$, where g^* was defined in equation (17).

Define

$$\begin{matrix} w_{\text{inv}}^{*} \\ w_{\text{inv}}^{0} \end{matrix} z_{\text{inv}} = \begin{matrix} 1 \\ 1 \end{matrix}. \tag{28}$$

There are two possibilities a) w_{inv}^{0} is not aligned with w_{inv}^{*} in which case the rank of the matrix in the above equation (28) is two and as a result the range space of the matrix spans all two-dimensional vectors, b) w_{inv}^{0} is aligned with w_{inv}^{*} but since $kw_{\text{inv}}^{0}k = 1$, $w_{\text{inv}}^{0} = \quad w_{\text{inv}}^{*}$ in which case $z_{\text{inv}} = w_{\text{inv}}^{*}$ solves the above equation (28). In both the cases the equation (28) has a solution. Let the solution of the above equation (28) be $\bar{z}_{\text{inv}}$. Define $x = S \quad (\bar{z}_{\text{inv}}; 0)$. Therefore, from equation (28) it follows that $g^*(x) \not\equiv g^{0}(x)$. See the simplification below for the justification.

$$\begin{aligned} g^*(x) &= I \quad (w_{\text{inv}}^{*}; 0) \quad S^{-1}(x) \quad = I(w_{\text{inv}}^{*} \quad \bar{z}_{\text{inv}}) = 1 \\ g^{0}(x) &= I \quad (w_{\text{inv}}^{0}; 0) \quad S^{-1}(x) \quad = I(w_{\text{inv}}^{0} \quad \bar{z}_{\text{inv}}) = 0 \end{aligned} \tag{29}$$

We showed above that $g^* \in W_{\text{inv}}^{\dagger}$ and $g^{0} \in W_{\text{inv}}^{\dagger}$ are two distinct functions. Recall in Lemma 4, we showed W_{inv} has infinitely many distinct hyperplanes. We can select any pair of hyperplanes W_{inv}, for the corresponding functions in the set $W_{\text{inv}}^{\dagger}$ the condition in equation (28) continues to hold. Thus we can conclude that there are infinitely many distinct functions in $W_{\text{inv}}^{\dagger}$.

Recall we described above that an algorithm can successfully solve equation (1), if and only if the output $f = g^*$. Observe that the same exact training data $fD^e g_{e\in\mathcal{E}_{tr}}$ can be generated by any other labelling hyperplane $w_{\text{inv}}^{0} \not\equiv w_{\text{inv}}^{*}$, where $(w_{\text{inv}}^{0}; 0) \in W_{\text{inv}}$ (this follows from the definition of W_{inv} in equation (16)). Define $g^{0} = I \quad (w^{0}; 0) \quad S^{-1}$, where $g^{0} \in W_{\text{inv}}^{\dagger}$. From the justification above, we

know that $g^0 \not\equiv g$. Since $g^0 \not\equiv g^*$ the algorithm can only be successful on one of the two labelling hyperplanes w^0_{inv} or w^*_{inv}. In fact, since we showed that there are infinitely many possible distinct hyperplanes in W_{inv}, the algorithm can only succeed on one of them. To summarize, the algorithm fails almost everywhere on the entire set, W_{inv}, of equivalent generating models. This completes the proof. □

Remark on extension under partially informative invariant features, i.e., Assumption 9. The impossibility result extends to the case when the environments follow Assumption 9. The first thing to note is that from Lemma 2, g^* continues to be the OOD optimal solution hyperplane. In the above proof, we had shown the construction of how there are infinitely many possible equally good hyperplanes that could have generated the data. To arrive at those hyperplanes, we relied on Lemma 3, where we showed that there are multiple candidate hyperplanes that could have generated the same training data. In the lemma, we only exploited the separability of latent invariant features and boundedness. If we continue to assume separability and boundedness for invariant features, then the result from Lemma 3 can be used in this case as well. As a result, we can continue to use the claim that there are multiple equally good candidate hyperplanes that the algorithm cannot distinguish. Thus the impossibility result extends to this setup too.

Remark on inveribility of S. The entire proof only requires us to assume to be able to have invertibility on the latent invariant features, i.e., we should be able to recover Z^e_{inv} from X^e. Therefore, Theorem 2 extends to matrices S that are only invertible upto the Z^e_{inv}.

Remark on impossibility under continuous random variable assumption. In the proof, we showed that if the test environment e places all the mass on the solution of equation (28), then the algorithm fails. In the setting, where we are only allowed to work with continuous random variables, can we continue to claim impossibility? The answer is yes. The reason is quite simple, we can instead of using the solution to equation (28) construct a small ball around that region. Since the solution to equation (28) that we constructed is in the interior of the half-spaces such an argument works.

Remark on multi-class classification. We describe a natural extension of the model in Assumption 2 to k-class classification.

Assumption 10. ***Linear classification structural equation model (FIIF) for multi-class classification.*** *In each* $e \in E_{all}$

$$\begin{aligned} Y^e &\quad \arg\max(W^*_{\text{inv}} \quad Z^e_{\text{inv}}) \\ X^e &\quad S \quad Z^e_{\text{inv}}; Z^e_{\text{spu}} \quad ; \end{aligned} \tag{30}$$

where $W^*_{\text{inv}} \in \mathrm{R}^{k \times m}$, $\arg\max$ *is taken over the* k *rows to generate the label* Y^e, $S \in \mathrm{R}^{d \times d}$.

We can add noise as well in the above SEM, which uniformly at random switches the class. The key geometric intuition for the impossibility result that we proved above, which was illustrated in Figure 1, carries over to this case provided the label generating hyperplane separates the supports of adjacent classes with a finite margin. Following the same geometric intuition, we can generalize the formal impossibility proof to this case as well for the SEM in Assumption 10.

A.5 Proof of Theorem 3: sufficiency and insufficiency of ERM and IRM

Lemma 4. *If Assumptions 2, 4, 7 hold, then there exists a classifier which puts a non-zero weight on the spurious feature and continues to be Bayes optimal in all the training environments.*

Proof of Lemma 4. We will follow the construction based on Lemma 3's proof.

Choose an arbitrary non-zero vector $\quad 2\ \mathsf{R}^o$. We will derive a bound on the margin of $(w^*_{\text{inv}};\ )$. Consider a $z_{\text{inv}}\ 2\ [_{e\in\mathcal{E}_{tr}} Z^e_{\text{inv}}$ and a $z_{\text{spu}}\ 2\ [_{e\in\mathcal{E}_{tr}} Z^e_{\text{spu}}$. Define $y^* = \text{sgn}(w^*_{\text{inv}}\ \ z_{\text{inv}})$. The margin $(w^*_{\text{inv}};\ )$ at this point $(z_{\text{inv}}; z_{\text{spu}})$ with respect to y^* is defined as

$$y^*\ \ w^*_{\text{inv}}\ \ z_{\text{inv}}\ \ + y^*\ \ \ \ z_{\text{spu}}\ : \tag{31}$$

Using Cauchy-Schwarz inequality, we get

$$jy^*\ \ \ \ z_{\text{spu}}\ j = j\ \ \ z_{\text{spu}}j\ \ \ k\ \ kkz_{\text{spu}}k: \tag{32}$$

Since the train support of spurious feature is bounded we can set the magnitude of $\quad$ sufficiently small to control $y^*\ \ \ \ z_{\text{spu}}\ .$ If $k\ \ k\ \ \ \frac{c}{2z^{\text{sup}}}$, then $j\ \ \ z_{\text{spu}}j\ \ \ \frac{c}{2}$, where z^{sup} satisfies the following condition – for each $z\ 2\ [_{e\in\mathcal{E}_{tr}} Z^e_{\text{spu}}$ and $kzk\ \ \ z^{\text{sup}}$. We can use this to find a bound on the margin as follows. Recall from equation (22) we have

$$y^*\ \ w^*_{\text{inv}}\ \ z_{\text{inv}}\ \ \ \ \ c: \tag{33}$$

We use the condition $j\ \ \ z_{\text{spu}}j\ \ \ \frac{c}{2}$ in the simplification below

$$y^*\ \ w^*_{\text{inv}}\ \ z_{\text{inv}}\ \ + y^*\ \ \ \ z_{\text{spu}}\ \ \ \ c\ \ \ j\ \ \ z_{\text{spu}}j\ \ \ \frac{c}{2}: \tag{34}$$

From the above equation it follows that $\text{sgn}\ (w^*_{\text{inv}};\ )\ (z_{\text{inv}}; z_{\text{spu}})\ = \text{sgn}\ (w^*_{\text{inv}}; 0)\ (z_{\text{inv}}; z_{\text{spu}})\ \ =)$ $\mathsf{I}\ (w^*_{\text{inv}};\ )\ (z_{\text{inv}}; z_{\text{spu}})\ = \mathsf{I}\ (w^*_{\text{inv}}; 0)\ (z_{\text{inv}}; z_{\text{spu}})$. This condition holds for each $z_{\text{inv}}\ 2\ [_{e\in\mathcal{E}_{tr}} Z^e_{\text{inv}}$ and a $z_{\text{spu}}\ 2\ [_{e\in\mathcal{E}_{tr}} Z^e_{\text{spu}}$. We use this condition to compute the error of a classifier based on $(w^*_{\text{inv}};\ )$ below. Define $g^*_{\text{spu}} = \mathsf{I}\ \ (w^*_{\text{inv}};\ )\ \ S^{-1}$. The error achieved by g^*_{spu} is

$$\begin{aligned} R^e(g^*_{\text{spu}}) &= \mathsf{E}\left[\ Y^e\ \ \mathsf{I}\ (w^*_{\text{inv}};\ )\ \ (z_{\text{inv}}; z_{\text{spu}})\right] \\ &= \mathsf{E}\left[\ \mathsf{I}\ (w^*_{\text{inv}}; 0)\ \ (z_{\text{inv}}; z_{\text{spu}})\ \ \ \ N^e\ \ \mathsf{I}\ (w^*_{\text{inv}};\ )\ \ (z_{\text{inv}}; z_{\text{spu}})\ \right]\ = \mathsf{E}\ N^e\ = q: \end{aligned} \tag{35}$$

The same calculation as above equation (35) holds in all the training environments. Thus g^*_{spu} achieves the minimum error possible q for all the training environments $e\ 2\ E_{tr}$. □

We restate Theorem 3 for convenience.

Theorem 7. ***Sufficiency and Insufficiency of ERM and IRM.*** *Suppose each $e\ 2\ E_{all}$ follows Assumption 2. Assume that a) the invariant features are strictly separable, bounded, and satisfy support overlap, b) the spurious features are bounded (Assumptions 3-5, 7 hold).*

Sufficiency: *If the spurious features satisfy support overlap (Assumption 6 holds), then both ERM and IRM solve the OOD generalization problem (eq.* (1)*). Also, there exist ERM and IRM solutions that rely on the spurious features and still achieve OOD generalization.*

Insufficiency: *If spurious features do not satisfy support overlap, then both ERM and IRM fail at solving the OOD generalization problem (eq.* (1)*). Also, there exist no such classifiers that rely on the spurious features and still achieve OOD generalization.*

Proof of Theorem 7. Let us begin with the first part of the Theorem. We first show that there exist solutions to ERM and IRM that rely on spurious features that also achieve OOD generalization (that is solve (1)). Since Assumptions 2, 4, 7, hold we can use Lemma 4. From Lemma 4, it follows that for each $z_{\text{inv}}\ 2\ [_{e\in\mathcal{E}_{tr}} Z^e_{\text{inv}}$ and for each $z_{\text{spu}}\ 2\ [_{e\in\mathcal{E}_{tr}} Z^e_{\text{inv}}$:

$$\mathsf{I}\ (w^*_{\text{inv}};\ )\ \ (z_{\text{inv}}; z_{\text{spu}})\ \ = \mathsf{I}\ (w^*_{\text{inv}}; 0)\ \ (z_{\text{inv}}; z_{\text{spu}})\ : \tag{36}$$

From Assumption 5 and 6 it follows that for each $z_{\text{inv}} \in \bigcap_{e \in \mathcal{E}_{all}} Z^e_{\text{inv}}$ and for each $z_{\text{spu}} \in \bigcap_{e \in \mathcal{E}_{all}} Z^e_{\text{inv}}$.

$$\Phi(w^*_{\text{inv}}; \cdot)(z_{\text{inv}}; z_{\text{spu}}) = \Phi(w^*_{\text{inv}}; 0)(z_{\text{inv}}; z_{\text{spu}}) \tag{37}$$

Therefore, the error of the classifier $g^*_{\text{spu}} = \Phi(w^*_{\text{inv}}; \cdot) \circ S^{-1}$ in each environment $e \in E_{all}$ is

$$\begin{aligned} R^e(g^*_{\text{spu}}) &= \mathrm{E}\left[Y^e \neq \Phi(w^*_{\text{inv}}; \cdot)(z_{\text{inv}}; z_{\text{spu}}) \right] \\ &= \mathrm{E}\left[\Phi(w^*_{\text{inv}}; 0)(z_{\text{inv}}; z_{\text{spu}}) \neq N^e \right] \neq \Phi(w^*_{\text{inv}}; \cdot)(z_{\text{inv}}; z_{\text{spu}}) \Big] = \mathrm{E}[N^e] = q. \end{aligned} \tag{38}$$

g^*_{spu} is Bayes optimal on each environment $e \in E_{all}$. Therefore, g^*_{spu} also solves equation (1). Since g^*_{spu} is optimal in all the environments, it also solves ERM as it also minimizes the sum of risks across training environments. g^*_{spu} is also a valid invariant predictor since it is simultaneously optimal across all the environments. Since g^*_{spu} achieves an average error of q across training environments, each solution to ERM and IRM has to achieve an error of q in all the training environments as well. Since the solution to ERM and IRM achieves an error of q it cannot differ from g^* at any point in the training support. This argument holds in a pointwise sense when Z^e_{inv} is a discrete random variable, otherwise, say when Z^e_{inv} is a continuous random variable this argument can only be violated over a set of measure zero.[15] Owing to the support overlap between E_{tr} and E_{all}, each solution to ERM and IRM continues to succeed in E_{all}. This completes the first part of the proof.

We now move to the next part of the theorem, where the spurious features do not satisfy support overlap assumption (Assumption 6). Consider a linear classifier that the method learns $\Phi \circ w$, where Φ is composed with a linear function. The classifier operates on x, and we get $\Phi(w \cdot x)$ and since $x = Sz$ (from Assumption 2) we can write this as $\Phi(w \cdot S(z))$. To simplify notation, we call $\Phi \circ w \circ S = \Phi \cdot w$. Our goal is to show that if w assigns a non-zero weight to the spurious features, then $\Phi \circ w \circ S$ cannot solve the OOD generalization problem (eq. (1)). We write $w = (w_{\text{inv}}; w_{\text{spu}})$. Suppose $w_{\text{spu}} \neq 0$ and yet the classifier solves the problem in equation (1). Consider the classifier that generates the data $(w^*_{\text{inv}}; 0)$. Pick any point $z_{\text{inv}} \in \bigcap_{e \in \mathcal{E}_{all}} Z^e_{\text{inv}}$ and pick any non-zero $z^e_{\text{spu}} \in \mathrm{R}^o$. Call $z = (z_{\text{inv}}; z_{\text{spu}})$ We divide the analysis into two cases.

Case 1: $\Phi(w_{\text{inv}}; w_{\text{spu}}) \cdot z \neq \Phi(w^*_{\text{inv}}; 0) \cdot z$. In this case, $(w_{\text{inv}}; w_{\text{spu}})$ cannot solve equation (1) as there exists a test environment where we have all the mass on z.

Case 2: $\Phi(w_{\text{inv}}; w_{\text{spu}}) \cdot z = \Phi(w^*_{\text{inv}}; 0) \cdot z$. Observe that since $w_{\text{spu}} \neq 0$, we can increase or decrease one of the components of z_{spu} corresponding to a non-zero w_{spu} until the two classifiers disagree in which case we get Case 1. Note that since Assumption 6 does not hold, we are allowed to change z_{spu} arbitrarily.

Thus we have established that a classifier cannot be OOD optimal if it assigns a non-zero weight to the spurious feature. As a result, the classifier from the first part g^*_{spu} which assigned non-zero weight to spurious features cannot be OOD optimal without the Assumption 6. However, g^*_{spu} continues to be in the solution space of both ERM and IRM as it is still Bayes optimal across all the train environments, which is why both ERM and IRM fail. At this point the proof of the statement of theorem is complete. However, we give a characterization of optimal solutions in the next paragraph.

Now let us consider any classifier in $w \in W_{\text{inv}}$ (from equation (16)) written as $w = (w_{\text{inv}}; 0)$. For such a classifier by definition it is true that for each $z_{\text{inv}} \in \bigcap_{e \in \mathcal{E}_{tr}} Z^e_{\text{inv}}$, $\Phi \cdot w_{\text{inv}} \cdot z_{\text{inv}} = \Phi \cdot w^*_{\text{inv}} \cdot z_{\text{inv}}$. From Assumption 5 it follows that for each $z_{\text{inv}} \in \bigcap_{e \in \mathcal{E}_{all}} Z^e_{\text{inv}}$, $\Phi \cdot w_{\text{inv}} \cdot z_{\text{inv}} = \Phi \cdot w^*_{\text{inv}} \cdot z_{\text{inv}}$ and thus the classifier continues to achieve an error of q on all the test environments. Thus we can conclude that $\Phi \circ w \circ S^{-1}$ is OOD optimal. Therefore, all the elements in the set $W^{\dagger}_{\text{inv}}$ (from eq. (27)) are OOD optimal.

□

Remark on invertibility of S. The proof extends to the case when we can invert and recover entire Z^e_{inv} and also recover at least one component of the spurious features Z^e_{spu}.

Remark on failure of ERM and IRM under continuous random variable assumption. In the proof, we showed that if the test environment e places all the mass on the solution to Case 1, then the

[15] The continuous random variable case can give rise to some pathological shifts. We show later in the proof of Theorem 4 as to why we do not need to worry about these pathological shifts.

algorithm fails. In the setting, where we are only allowed to work with continuous random variables, can we continue to make the claim for impossibility? The answer is yes. The reason is quite simple, we can instead of using the solution to Case 1 construct a small ball around that region, where the classifiers continue to disagree.

Remark on multi-class classification. We extend the result to the above SEM in Assumption 10. The reason ERM and IRM fail in this case is two fold – a) there exists a hyperplane that perfectly separates the support of the invariant features with a finite margin and b) support of spurious features are allowed to change. In the multi-class case, we can use the same reasoning – if there is a hyperplane that perfectly separates for adjacent classes, ERM and IRM continue to fail as long as the support of spurious features is allowed to change.

A.6 Proof of Theorem 4: IB-IRM and IB-ERM vs. IRM and ERM

We now lay down some properties of the entropy of discrete random variables and in parallel also lay down the properties of differential entropy of continuous random variables. Recall that a discrete random variable has a non-zero probability at each point in its support and a continuous random variable has a zero probability (and a positive density) at each point in the support.

The entropy or the Shannon entropy of a discrete random variable $X \sim P_X$ with support $\mathcal{X}$ is defined as

$$H(X) = \sum_{x \in \mathcal{X}} P_X(X = x) \log \frac{1}{P_X(X = x)}. \tag{39}$$

The differential entropy of a continuous random variable $X \sim P_X$ with support $\mathcal{X}$ is given as follows

$$h(X) = \int_{x \in \mathcal{X}} \log \frac{1}{dP_X(x)} \, dP_X(x), \tag{40}$$

where $dP_X(x)$ is the Radon-Nikodym derivative of P_X w.r.t the Lesbegue measure.

Lemma 5. *If X and Y are discrete scalar valued random variables that are independent, then*

$$H(X + Y) \geq \max\left\{H(X), H(Y)\right\}.$$

Proof of Lemma 5. Define $Z = X + Y$.

$$\begin{aligned}
H(Z|X) &= \sum_{x \in \mathcal{X}} P_X(x) \sum_{z \in \mathcal{Z}} P_{Z|X}(Z = z|X = x) \log \frac{1}{P_{Z|X}(Z = z|X = x)} \\
&= \sum_{x \in \mathcal{X}} P_X(x) \sum_{z \in \mathcal{Z}} P_{Y|X}(Y = z - x|X = x) \log \frac{1}{P_{Y|X}(Y = z - x|X = x)} \\
&= \sum_{x \in \mathcal{X}} P_X(x) \sum_{z \in \mathcal{Z}} P_{Y|X}(Y = z - x|X = x) \log \frac{1}{P_{Y|X}(Y = z - x|X = x)} \quad (\text{use } X \perp Y) \\
&= \sum_{x \in \mathcal{X}} P_X(x) \sum_{z \in \mathcal{Z}} P_Y(Y = z - x) \log \frac{1}{P_Y(Y = z - x)} \\
&= H(Y)
\end{aligned} \tag{41}$$

$$\begin{aligned}
I(Z; X) &= H(Z) - H(Z|X) = H(X + Y) - H(Y) \\
I(Z; Y) &= H(Z) - H(Z|Y) = H(X + Y) - H(X)
\end{aligned} \tag{42}$$

From equation (42) and the property of mutual information that $I(Z; X) \geq 0; I(Z; Y) \geq 0$ it follows that

$$H(X + Y) \geq H(Y); \; H(X + Y) \geq H(X) \implies H(X + Y) \geq \max\{H(X), H(Y)\}. \tag{43}$$

This completes the proof. □

Lemma 6. *If X and Y are continuous scalar valued random variables that are independent, then*

$$h(X + Y) \geq \max\left\{h(X), h(Y)\right\}.$$

Proof of Lemma 6. Define $Z = X + Y$.

$$\begin{aligned}
h(Z|X) &= \mathbb{E}_{P_X}\left[\mathbb{E}_{P_{Z|X}}\left[\log \frac{1}{dP_{Z|X}(Z = z|X = x)}\right]\right] \\
&= \mathbb{E}_{P_X}\left[\mathbb{E}_{P_{Y|X}}\left[\log \frac{1}{dP_{Y|X}(Y = z - x|X = x)}\right]\right] \quad (\text{use } X \perp Y) \\
&= h(Y)
\end{aligned} \tag{44}$$

Note that $I(Z;X) \geq 0 \implies h(Z) \geq h(Z|X)$. Combining this with the above equation (44) we get

$$h(X+Y) \geq h(Y). \tag{45}$$

From symmetry it follows that $h(X+Y) \geq h(X)$. This completes the proof. □

Lemma 7. *If X and Y are discrete scalar valued random variables that are independent with the supports satisfying $2 \leq |\mathcal{X}| < \infty$, $2 \leq |\mathcal{Y}| < \infty$, then*

$$H(X+Y) > \max\left\{H(X), H(Y)\right\}.$$

Proof of Lemma 7. Suppose $\mathcal{X} = \{x_{\min}, \ldots, x_{\max}\}$ and $\mathcal{Y} = \{y_{\min}, \ldots, y_{\max}\}$. The smallest value of $X+Y$ is $x_{\min}+y_{\min}$ and the largest value is $x_{\max}+y_{\max}$. Suppose that the inequality in the claim is not true in which case from Lemma 5 it follows $H(X+Y) = H(X)$ or $H(X+Y) = H(Y)$. Suppose $H(X+Y) = H(X)$, then from equation (42) it follows that $I(X+Y;Y) = 0 \implies X+Y \perp Y$. Observe that if $Z = x_{\max}+y_{\max} \implies Y = y_{\max}$. Therefore, $\mathsf{P}(Y = y_{\max}|Z = x_{\max}+y_{\max}) = 1$. However, $\mathsf{P}(Y = y_{\max}) \neq 1$ as the support of Y has at least two elements. This contradicts $X+Y \perp Y$. As a result, $H(X+Y) \neq H(X)$. We can symmetrically show that $H(X+Y) \neq H(Y)$. Combining this with Lemma 5, it follows that $H(X+Y) > \max\{H(X), H(Y)\}$. □

Lemma 8. *If X and Y are continuous scalar valued random variables that are independent and have a bounded support, then*

$$h(X+Y) > \max\left\{h(X), h(Y)\right\}$$

Proof of Lemma 8. The steps of the proof are similar to Lemma 7. Suppose the inequality in the claim is not true in which case from Lemma 6 it follows that either $h(X+Y) = h(X)$ or $h(X+Y) = h(Y)$. Suppose $h(X+Y) = h(X)$ which implies $I(X+Y;Y) = 0 \implies X+Y \perp Y$. The support of X can be written in the form of union of intervals. Suppose we consider the rightmost interval and we write it as $[x_{\max}-\epsilon, x_{\max}]$. Similarly for Y, we write the rightmost interval as $[y_{\max}-\epsilon, y_{\max}]$. [16] Define an event $M : x_{\max}+y_{\max}-\epsilon \leq X+Y \leq x_{\max}+y_{\max}$. If M occurs, then $Y \geq y_{\max}-\epsilon$ and $X \geq x_{\max}-\epsilon$.

$$\begin{aligned} \mathsf{P}_X(X \geq x_{\max}-\epsilon \,|\, M) &= 0 \\ \mathsf{P}_Y(Y \geq y_{\max}-\epsilon \,|\, M) &= 0 \end{aligned} \tag{46}$$

If $\epsilon < 0$ we know that

$$\begin{aligned} \mathsf{P}_X(X \geq x_{\max}-\epsilon) &> 0 \\ \mathsf{P}_Y(Y \geq y_{\max}-\epsilon) &> 0 \end{aligned} \tag{47}$$

If $X+Y \perp Y$ then $\mathsf{P}_Y(Y \geq y_{\max}-\epsilon) = \mathsf{P}_Y(Y \geq y_{\max}-\epsilon \,|\, M)$, which is not the case from the above equations (46) and (47). Thus $X+Y \not\perp Y \implies I(X+Y;Y) > 0 \implies h(X+Y) > h(X)$. We can say the same for Y and conclude that $h(X+Y) > h(Y)$. This completes the proof. □

Theorem 4 has two versions – one for discrete random variables, and the other for continuous random variables. We discuss the discrete random variable case first as its easier to understand and then move to the continuous random variable case.

A.6.1 Discrete random variables

In this section, we assume that in each $e \in \mathcal{E}_{all}$, random variables $Z^e_{\text{inv}}, Z^e_{\text{spu}}, N^e, W^e$ in Assumption 8 are discrete. We formulate the optimization in terms of Shannon entropy as follows.

$$\begin{aligned} &\min_{w \in \mathbb{R}^k,\ r,\ \in \mathbb{R}^r,\ d} \frac{1}{|\mathcal{E}_{tr}|} \sum_e H^e(w, \quad) \\ &\text{s.t.} \ \frac{1}{|\mathcal{E}_{tr}|} \sum_e R^e(w, \quad) \leq r^* \\ &\quad w \in \arg\min_{w \in \mathbb{R}^k,\ r} R^e(w, \quad) \end{aligned} \tag{48}$$

Note that the only difference between equation (48) and the equation (6) is that the objective here is Shannnon entropy, while the objective in the other case is the differential entropy.

[16] We use same ϵ for both X and Y because can take the smaller of the rightmost intervals for X and Y.

Theorem 8. ***IB-IRM and IB-ERM vs IRM and ERM***

Fully informative invariant features (FIIF). *Suppose each $e \in E_{all}$ follows Assumption 2. Assume that the invariant features are strictly separable, bounded, and satisfy support overlap (Assumptions 3,5 and 7 hold). Also, for each $e \in E_{tr}$ $Z^e_{spu} \quad AZ^e_{inv} + W^e$, where $A \in \mathbb{R}^{o \times m}$, $W^e \in \mathbb{R}^o$ is discrete, bounded noise, with zero mean (and each component takes at least two distinct values). Each solution to IB-IRM (eq.* (6)*, with ℓ as* 0-1 *loss, and $r^{th} = q$), and IB-ERM solves the OOD generalization (eq.* (1)*) but ERM and IRM (eq.*(3)*) fail.*

In the above Theorem 8, we only state the first part of the Theorem 4, the reason is that the proof of the second part proof is exactly the same in both discrete and continuous random variable case and we describe the proof for the second part in the continuous random variable section next.

Proof of Theorem 8. First, let us discuss why IRM and ERM fail in the above setting. We argue that the failure, in this case, follows directly from the second part of Theorem 3. To directly use the second part of Theorem 3, we need Assumptions 2-5 and 7 to hold. In the statement of the above theorem, Assumption 2, 3, 5, and 7 already hold. We are only required to show that Assumption 4 holds. Since Z^e_{inv} and W^e are bounded on training environments we can argue that Z^e_{spu} is also bounded in training environments ($\|Z^e_{spu}\| \quad \|A\|\|Z^e_{inv}\| + \|W^e\|$). We can now directly use the second part of Theorem 3 because Assumptions 2-5 and 7 hold. Since Assumption 6 is not required to hold, both ERM and IRM will fail as their solution space continue to contain classifiers that rely on spurious features. To further elaborate on why ERM and IRM fail, recall that in the second part of Theorem 3, we relied on Lemma 4. In Lemma 4, we had shown that if latent invariant features are strictly separable, and latent spurious features are bounded, then there exist classifiers that rely on spurious features and yet are Bayes optimal on all the training environments. In this case, we have latent invariant features that are strictly separable and spurious features that are bounded, which is why we can use Theorem 3. We now move to the part, where we establish why IB-IRM and IB-ERM succeed.

Consider a solution to equation (48) and call it †. Consider the prediction made by this model

$$^\dagger X^e = \ ^\dagger S(Z^e_{inv}; Z^e_{spu}) = \ _{inv} \ Z^e_{inv} + \ _{spu} \ Z^e_{spu}. \tag{49}$$

We first show that $_{spu}$ is zero. We prove this by contradiction. Assume $_{spu} \neq 0$ and use the condition in the theorem to simplify the expression for the prediction as follows

$$\begin{aligned} & _{inv} \ Z^e_{inv} + \ _{spu} \ Z^e_{spu} \\ &= \ _{inv} \ Z^e_{inv} + \ _{spu} \ (AZ^e_{inv} + W^e) \\ &= \ _{inv} \ Z^e_{inv} + \ _{spu} \ (AZ^e_{inv} + W^e) \\ &= \ _{inv} + \ _{spu} \ A \ Z^e_{inv} + \ _{spu} \ W^e. \end{aligned} \tag{50}$$

We will show that $^+ = \ _{inv} + \ _{spu} \ A \ ; 0 \ S^{-1} = \ _{inv} + \ _{spu} \ A \ S^\dagger_{inv}$, where $S^\dagger_{inv}$ corresponds to the first m rows of the matrix S^{-1}, can continue to achieve an error of q and has a lower entropy than †. Recall that † achieves an average error across the training environments of q (because $r^{th} = q$ the average cannot fall below q as in that case at least one environment would have a lower error than q which is not possible), which implies each environment also achieves an error of q.

Consider an environment $e \in E_{tr}$. Since the error † is q it implies that for each training environment e

$$\ell(w^*_{inv} \ Z^e_{inv}) = \ell(\ _{inv} \ Z^e_{inv} + \ _{spu} \ Z^e_{spu}) \tag{51}$$

holds over all the points in the support of environment e. Suppose the above claim was not true, i.e. suppose the set $\ell(w^*_{inv} \ Z^e_{inv}) \neq \ell(\ _{inv} \ Z^e_{inv} + \ _{spu} \ Z^e_{spu})$ occurs with a for some point in the support (suppose that point occurs with probability). Let us compute the error

$$\begin{aligned} R^e(\ ^\dagger) &= \mathbb{E}\left[\ell(w^*_{inv} \ Z^e_{inv}) \quad N^e \quad \ell(\ _{inv} \ Z^e_{inv} + \ _{spu} \ Z^e_{spu})\right] \\ &= \ \mathbb{E}[1 \quad N^e] + (1 \quad)\mathbb{E}[N^e] > q \end{aligned} \tag{52}$$

If the above is true, then that contradicts the claim that † achieves an error of q. Thus the statement in equation (51) has to hold at all points in the training support of the invariant features. Let W^e be the support of W^e. In each training environment, if we consider a z^e_{inv} 2 Z^e_{inv}, then $8w^e$ 2 W^e, the following holds – if $\mathsf{I}(w^*_{\text{inv}} \quad z^e_{\text{inv}}) = 1$, then

$$\begin{aligned}
& {}_{\text{inv}} \; z^e_{\text{inv}} + {}_{\text{spu}} \; (Az^e_{\text{inv}} + w^e) \quad 0 \\
=) \; & {}_{\text{inv}} \; z^e_{\text{inv}} + {}_{\text{spu}} \; (Az^e_{\text{inv}}) \qquad {}_{\text{spu}} \; w^e \\
=) \; & {}_{\text{inv}} + {}_{\text{spu}} \; A \quad z^e_{\text{inv}} \quad \max_{w^e \in \mathcal{V}^e} \qquad {}_{\text{spu}} \; w^e \\
=) \; & {}_{\text{inv}} + {}_{\text{spu}} \; A \quad z^e_{\text{inv}} \quad 0 \\
=) \; & {}^{+} X^e \quad 0:
\end{aligned} \tag{53}$$

Similarly, we can argue that if $\mathsf{I}(w^*_{\text{inv}} \quad z^e_{\text{inv}}) = 0$, then

$$\begin{aligned}
& {}_{\text{inv}} + {}_{\text{spu}} \; A \quad z^e_{\text{inv}} < 0 \\
& {}^{+} X^e < 0:
\end{aligned} \tag{54}$$

In the above simplification equation (53), we use $\max_{w^e} \; {}_{\text{spu}} \; w^e \quad 0$. Consider any component of ${}_{\text{spu}}$; if the sign of the component is positive (negative), then set the corresponding component of w^e to be positive (negative). As a result, ${}_{\text{spu}} \; w^e \quad 0$. In this argument, we only relied on the assumption that w^e can take both signs in the set W^e. Suppose W^e had either positive or negative values only then this would imply that the mean of w^e is strictly positive or negative, which cannot be true because W^e is zero mean. From equation (53) and (54), we can conclude that $^{+}$ achieves the same error of q in all the training environments.

Observe that we can write $^{\dagger} \; X^e = {}^{+} \; X^e + {}_{\text{spu}} \; W^e$. We state two properties that we use to show that entropy $^{+}$ is smaller than †:

a) ${}_{\text{spu}} \; W^e \; ? \; {}^{+} \; X^e \; (\; {}^{+} \; X^e = \left[{}_{\text{inv}} + {}_{\text{spu}} \; A \right] \; Z^e_{\text{inv}}$ and $Z^e_{\text{inv}} \; ? \; W^e)$,

b) ${}^{+} \; X, \; {}_{\text{spu}} \; W^e$ are discrete random variables with finite support of size at least two.

We justify why b) is true in the above. ${}^{+} \; X^e$ is a bounded random variable (Z^e_{spu} is bounded as Z^e_{inv} and W^e are bounded. Thus X^e is also bounded). ${}^{+} \; X^e$ has at least two elements in its support this follows from equation (53) and (54). ${}_{\text{spu}} \; W^e$ is bounded since W^e is bounded and takes at least two values because each component of W^e takes at least two distinct values.

From a), b), and Lemma 7 it follows that ${}^{+} \; X^e$ is a classifier with lower entropy. We already established that $^{+}$ achieves the same error as † for all the training environments. $^{+}$ achieves an error of q for all the training environments simultaneously. Since q is the smallest value for the error that is achievable, the invariance constraint in equation (71) is automatically satisfied. Therefore, $^{+}$ is strictly preferable to †. Thus the solution † cannot rely on the spurious features and ${}_{\text{spu}} = 0$.

Thus any solution † to equation (48) has to satisfy $^{\dagger} \; S = (\; {}_{\text{inv}}; 0)$ and $^{\dagger} \; S$ also satisfies

$$\mathsf{I}(w^*_{\text{inv}} \quad Z^e_{\text{inv}}) = \mathsf{I}(\; {}_{\text{inv}} \quad Z^e_{\text{inv}}): \tag{55}$$

Recall that in the second part of Theorem 3's proof we showed that if a solution does not rely on spurious features and satisfies equation (65) for all the points in the support, then under the support overlap assumptions such a solution is OOD optimal as well. Since we assume support overlap assumption holds for the invariant features, we use the same argument from the second part of Theorem 3 and it follows that the solution to equation (48) also solves equation (1). □

A.6.2 Continuous random variables

In this section, we assume that in each e 2 E_{all}, the random variables $Z^e_{\text{inv}}; Z^e_{\text{spu}}; N^e; W^e$ in Assumption 2 are continuous.

Lower bounding the differential entropy objective: In general, the differential entropy can be unbounded below. Following the work of Kirsch et al. (2020), we add an independent noise term to

the predictor to ensure that the entropy is lower bounded. Suppose w is the output of the predictor and the entropy of the predictor for the data in environment e as $h^e(w)$. Consider a prediction made by the classifier w (X^e); we add noise ^{e} (continuous, bounded random variable with a finite entropy) to this prediction to get w $(X) + ^e$. The differential entropy after noise addition as $h^e(w$ $(X^e) + ^e)$. Observe that $h^e(w$ $(X^e) + ^e)$ $h(^e)$. In the rest of the discussion, we just write $h^e(w$ $(X^e) + ^e)$ as $h^e(w)$ to make the notation less cumbersome. We constrain H (H_w) in the optimization in equation (6) to a set H $= f$ 2 $\mathsf{R}^{r\times d}$ j $0 <$ $_{\inf}$ k k $_{\sup}g$ $(H_w = fw\ 2\ \mathsf{R}^{k\times r}\ j\ 0 < w_{\inf}$ kwk $w_{\sup}g)$ instead of H $= \mathsf{R}^{r\times d}$ $(H_w = \mathsf{R}^{k\times r})$. The reason to do this is that while the 0-1 loss does not change with scaling of the predictor but the entropy can change a lot. The lower bound on the norm of the classifier ensures that the optimization does not shrink it to zero in trying to minimize the entropy. We restate the optimization in equation (6) after accounting for the pathologies of differential entropy that we described above:

$$\begin{aligned}
&\min_{w\in H_w;\ \in H} \frac{1}{jE_{tr}j} \sum_e h^e\ w\quad)\\
&\text{s.t. } \frac{1}{jE_{tr}j} \sum_e R^e\ w\qquad r^{\text{th}}\\
&\quad w\ 2\ \arg\min_{w\in H_w} R^e(w\quad)
\end{aligned} \tag{56}$$

We restate Theorem 4 for convenience.

Theorem 9. ***IB-IRM and IB-ERM vs IRM and ERM***

Fully informative invariant features (FIIF). *Suppose each e 2 E_{all} follows Assumption 2. Assume that the invariant features are strictly separable, bounded, and satisfy support overlap (Assumptions 3,5 and 7 hold). Also, for each e 2 E_{tr} Z^e_{spu} $AZ^e_{\text{inv}} + W^e$, where A 2 $\mathsf{R}^{o\times m}$, W^e 2 R^o is continuous, bounded, and zero mean noise. Each solution to IB-IRM (eq.* (6)*, with $\grave{}$ as* 0-1 *loss, and $r^{\text{th}} = q$), and IB-ERM solves the OOD generalization (eq.* (1)*) but ERM and IRM (eq.*(3)*) fail.*

Partially informative invariant features (PIIF). *Suppose each e 2 E_{all} follows Assumption 1 and 9 e 2 E_{tr} such that $\mathsf{E}[\ ^eZ^e_{\text{spu}}]$ $\neq$ 0. If $jE_{tr}j > 2d$ and the set E_{tr} lies in a linear general position (a mild condition defined in the Appendix), then each solution to IB-IRM (eq.* (6)*, with $\grave{}$ as square loss, $^2 < r^{\text{th}}$ ^2_Y, where ^2_Y and ^{2} are the variance in the label and noise across E_{tr}) and IRM (eq.*(3)*) solves OOD generalization (eq.* (1)*) but IB-ERM and ERM fail.*

Proof of Theorem 9. First, let us discuss why IRM and ERM fail in the above setting. We argue that the failure, in this case, follows directly from the second part of Theorem 3. To directly use the second part of Theorem 3, we need Assumptions 2-5 and 7 to hold. In the statement of the above theorem, Assumption 2, 3, 5, and 7 already hold. We are only required to show that Assumption 4 holds. Since Z^e_{inv} and W^e are bounded on training environments we can argue that Z^e_{spu} is also bounded in training environments ($kZ^e_{\text{spu}}k$ $kAkZ^e_{\text{inv}}k + kW^ek$). We can now directly use the second part of Theorem 3 because Assumptions 2-5 and 7 hold. Since Assumption 6 is not required to hold, both ERM and IRM will fail as their solution space continue to contain classifiers that rely on spurious features. [17]

Consider a solution to IB-IRM (eq. (56)) and call it †. Consider the prediction made by this model

$$^\dagger\ X^e = {}^\dagger\ S(Z^e_{\text{inv}}; Z^e_{\text{spu}}) = {}_{\text{inv}}\ Z^e_{\text{inv}} + {}_{\text{spu}}\ Z^e_{\text{spu}}. \tag{57}$$

We first show that $_{\text{spu}}$ is zero. We prove this by contradiction. Assume $_{\text{spu}}$ $\neq$ 0 and use the condition in the theorem to simplify the expression for the prediction as follows.

$$\begin{aligned}
&{}_{\text{inv}}\ Z^e_{\text{inv}} + {}_{\text{spu}}\ Z^e_{\text{spu}}\\
&= {}_{\text{inv}}\ Z^e_{\text{inv}} + {}_{\text{spu}}\ (AZ^e_{\text{inv}} + W^e)\\
&= \big({}_{\text{inv}}\ Z^e_{\text{inv}} + {}_{\text{spu}}\ (AZ^e_{\text{inv}} + W^e)\big)\\
&= {}_{\text{inv}} + {}_{\text{spu}}\ A\quad Z^e_{\text{inv}} + {}_{\text{spu}}\ W^e.
\end{aligned} \tag{58}$$

[17] In the remark following the proof of Theorem 3, we had discussed the failure of ERM and IRM continues to hold even when we are restricted to use continuous random variables.

We will show that $^{+} = \begin{bmatrix} _{\text{inv}} + _{\text{spu}} A & ; 0 \end{bmatrix} S^{-1} = \begin{bmatrix} _{\text{inv}} + _{\text{spu}} A \end{bmatrix} S^{\dagger}_{\text{inv}}$, where $S^{\dagger}_{\text{inv}}$ corresponds to the first m rows of the matrix S^{-1}, can continue to achieve an error of q and has a lower entropy than †. Recall that † achieves an average error across the training environments of q (because $r^{\text{th}} = q$ the average cannot fall below q as in that case at least one environment would have a lower error than q which is not possible), which implies each environment also achieves an error of q.

Consider an environment $e \in E_{tr}$. Since the error † is q it implies that for each training environment

$$\mathsf{I}(w^{*}_{\text{inv}} \quad Z^{e}_{\text{inv}}) = \mathsf{I}(\ _{\text{inv}} \quad Z^{e}_{\text{inv}} + \ _{\text{spu}} \quad Z^{e}_{\text{spu}}); \tag{59}$$

holds with probability 1. Suppose the above claim was not true, i.e. suppose the set $\mathsf{I}(w^{*}_{\text{inv}} \quad Z^{e}_{\text{inv}}) \neq \mathsf{I}(\ _{\text{inv}} \quad Z^{e}_{\text{inv}} + \ _{\text{spu}} \quad Z^{e}_{\text{spu}})$ occurs with a non-zero probability say $\quad$. Let us compute the error

$$\begin{aligned} R^{e}(\ ^{\dagger}) &= \mathsf{E}\left[\mathsf{I}(w^{*}_{\text{inv}} \quad Z^{e}_{\text{inv}}) \quad N^{e} \quad \mathsf{I}(\ _{\text{inv}} \quad Z^{e}_{\text{inv}} + \ _{\text{spu}} \quad Z^{e}_{\text{spu}})\right] \\ &= \quad \mathsf{E}[1 \quad N^{e}] + (1 \quad)\mathsf{E}[N^{e}] > q \end{aligned} \tag{60}$$

If the above is true, then that contradicts the claim that † achieves an error of q. Thus the statement in equation (59) has to hold with probability 1. Let W^{e} denote the support of W^{e} in environment e. We can restate the above observation as – there exists sets $\tilde{Z}^{e}_{\text{inv}} \quad Z^{e}_{\text{inv}}$ and a set $\tilde{W}^{e} \quad W^{e}$ such that $\mathsf{P}(\tilde{Z}^{e}_{\text{inv}} \quad \tilde{W}^{e}) = 1$ [18] and for each element in $\tilde{Z}^{e}_{\text{inv}} \quad \tilde{W}^{e}$

$$\mathsf{I}(w^{*}_{\text{inv}} \quad Z^{e}_{\text{inv}}) = \mathsf{I}(\ _{\text{inv}} \quad Z^{e}_{\text{inv}} + \ _{\text{spu}} \quad Z^{e}_{\text{spu}}) \tag{61}$$

Consider a training environment $e \in E_{tr}$. For each $z^{e}_{\text{inv}} \in \tilde{Z}^{e}_{\text{inv}}$, the following conditions hold $\forall w^{e} \in \tilde{W}^{e}$ – if $\mathsf{I}(w^{*}_{\text{inv}} \quad z^{e}_{\text{inv}}) = 1$, then

$$\begin{aligned} & _{\text{inv}} \quad z^{e}_{\text{inv}} + \ _{\text{spu}} \quad (Az^{e}_{\text{inv}} + w^{e}) \quad 0 \\ \Rightarrow & \ _{\text{inv}} \quad z^{e}_{\text{inv}} + \ _{\text{spu}} \quad (Az^{e}_{\text{inv}}) \quad \ _{\text{spu}} \quad w^{e} \\ \Rightarrow & \ _{\text{inv}} + \ _{\text{spu}} \quad A \quad z^{e}_{\text{inv}} \quad \max_{w^{e} \in \tilde{W}^{e}} \ _{\text{spu}} \quad w^{e} \\ \Rightarrow & \ _{\text{inv}} + \ _{\text{spu}} \quad A \quad z^{e}_{\text{inv}} \quad 0 \\ \Rightarrow & \ ^{+} X^{e} \quad 0. \end{aligned} \tag{62}$$

Similarly, we can argue that if $\mathsf{I}(w^{*}_{\text{inv}} \quad z^{e}_{\text{inv}}) = 0$, then

$$\begin{aligned} & _{\text{inv}} + \ _{\text{spu}} \quad A \quad z^{e}_{\text{inv}} < 0 \\ & ^{+} X^{e} < 0. \end{aligned} \tag{63}$$

In the above simplification in equation (62), we use $\max_{w^{e}} \ _{\text{spu}} \quad w^{e} \quad 0$. Consider any component of $_{\text{spu}}$; if the sign of the component is positive (negative), then set the corresponding component of w^{e} to be positive (negative). As a result, $_{\text{spu}} \quad w^{e} \quad 0$. In this argument, we only relied on the assumption that w^{e} can take both signs in the set $\tilde{W}^{e}$. Suppose w^{e} can only take either positive or negative values in $\tilde{W}^{e}$ this would imply that the mean of w^{e} is strictly positive or negative, which cannot be true because W^{e} is zero mean. From equation (62), (63), and $\mathsf{P}(\tilde{Z}^{e}_{\text{inv}} \quad \tilde{W}^{e}) = 1$, we can conclude that $^{+}$ achieves the same error of q in all the training environments.

Observe that we can write $^{\dagger} \quad X^{e} = \ ^{+} \quad X^{e} + \ _{\text{spu}} \quad W^{e}$. We state two properties that we use to show that entropy $^{+}$ is smaller than †:

a) $_{\text{spu}} \quad W^{e} \perp \ ^{+} \quad X^{e}$ ($^{+} \quad X^{e} = \begin{bmatrix} _{\text{inv}} + _{\text{spu}} A \end{bmatrix} Z^{e}_{\text{inv}}$ and $Z^{e}_{\text{inv}} \perp W^{e}$),

b) $^{+}_{\text{inv}} \quad X$, $_{\text{spu}} \quad W^{e}$ are continuous bounded random variables,

We justify why b) is true in the above. $^{+}_{\text{inv}} \quad X^{e}$ is a bounded random variable (Z^{e}_{spu} is bounded as Z^{e}_{inv} is bounded and as a result X^{e} is bounded as well). Observe that $^{+}_{\text{inv}} \neq 0$, this follows

[18] Owing to the independence of the noise we also have $\mathsf{P}(\tilde{Z}^{e}_{\text{inv}}) = 1, \mathsf{P}(\tilde{W}^{e}) = 1$.

from equation (62) and (63). $\Phi^{+}_{\text{inv}} X^e$ is a continuous random variable as well. Suppose $\Phi^{+}_{\text{inv}} X^e$ was not continuous, which implies for some constant b, $\Phi^{+}_{\text{inv}} X^e = b$ with a finite probability. If $\Phi^{+}_{\text{inv}} X^e = b$ with a finite probability, then X cannot be a continuous random vector (as there exists a hyperplane which occurs with a non-zero probability).

From a), b), and Lemma 8 it follows that

$$h^e(\Phi^{+} X^e) < h^e(\Phi^{\dagger} X^e) \tag{64}$$

Note that the above equation (64) is true independent of whether we added a bounded noise to keep the entropy bounded from below. Therefore, so far we have established that Φ^{+} is a classifier with lower entropy and the same error as $\Phi^{\dagger}$. Observe that Φ^{+} achieves an error of q for all the training environments simultaneously. Since q is the smallest value for the error that is achievable, the invariance constraint in equation (71) is automatically satisfied with $\Phi^{\dagger}$ as the classifier and the representation as the identity. Thus Φ^{+} is a strictly preferable solution $\Phi^{\dagger}$, which contradicts the optimality of $\Phi^{\dagger}$. Therefore, it follows that $\Phi_{\text{spu}} = 0$

Thus any solution $\Phi^{\dagger}$ to equation (56) has to satisfy $\Phi^{\dagger} S = (\Phi_{\text{inv}}; 0)$ and $\Phi^{\dagger} S$ also satisfies

$$\mathsf{I}(w^*_{\text{inv}} Z^e_{\text{inv}}) = \mathsf{I}(\Phi_{\text{inv}} Z^e_{\text{inv}}) \tag{65}$$

with probability one. From the second part of Theorem 3's proof we know if a solution satisfies two properties a) does not rely on spurious features, and b) satisfies equation (65) for all the points in the support, then under the support overlap of invariant features such a solution is OOD optimal (solves equation (1)) as well. In this case, we have also assumed support overlap assumption holds for the invariant features. We have established that the solution does not rely on spurious features. Also, we have shown that equation (65) holds not pointwise but with probability one. We can still use the same argument from the second part of Theorem 3 and it follows that the solution to equation (56) also solves equation (1). Next, we show why it suffices for the equation (65) to hold with probability one.

Since the equation (65) does not hold pointwise at all the points in the support and can be violated over a set of probability zero we need to be careful about some pathological shifts at test time that place a finite mass in the region where equation (1) is violated. We now argue using arguments based on standard measure theory (Ash and Doléans-Dade, 2000) that such pathological shifts cannot occur under the assumptions made in this setting.

Recall that we defined $Z^e_{\text{inv}} W^e$ to be the set where equation (65) holds pointwise. $\mathsf{P}(Z^e_{\text{inv}} W^e) = 1$. Owing to the independence $Z^e ? W^e$, we have $\mathsf{P}(Z^e_{\text{inv}}) = 1, \mathsf{P}(W^e) = 1$. It can be shown that the Lebesgue measure of the set $Z^e_{\text{inv}} \cap Z^e_{\text{inv}}$ is zero, i.e., $(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) = 0$. If the Lebesgue measure was positive, i.e., $(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) > 0$, then the probability of this set would also be non-zero, i.e., $\mathsf{P}(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) > 0$. The main insight to show this follows from the observation that the probability density is positive on the set $Z^e_{\text{inv}} \cap Z^e_{\text{inv}}$ since the set is part of the support of Z^e_{inv}.

A formal argument to show $(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) > 0 \implies \mathsf{P}(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) > 0$ goes as follows.

Assume the contrary, i.e., $\mathsf{P}(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) = 0$. Let the density be denoted as $f_{Z^e_{\text{inv}}}$. Define the set $P_k = \{z_{\text{inv}} \in Z^e_{\text{inv}} \cap Z^e_{\text{inv}} \mid f_{Z^e_{\text{inv}}}(z) > \frac{1}{k}\}$.

$$Z^e_{\text{inv}} \cap Z^e_{\text{inv}} = \bigcup_{k=1}^{\infty} P_k \tag{66}$$

$P_k \subseteq Z^e_{\text{inv}} \cap Z^e_{\text{inv}} \implies (P_k) \leq (Z^e_{\text{inv}} \cap Z^e_{\text{inv}})$. Since $(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) > 0$, $\exists$ some s for which $(P_s) > 0$.

Define g_s

$$g_s(x) = \begin{cases} \frac{1}{s} \text{ if } x \in P_k \\ 0 \text{ otherwise} \end{cases} \tag{67}$$

$$\mathsf{P}(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) = \int_{Z^e_{\text{inv}} \setminus Z^e_{\text{inv}}} f_{Z^e_{\text{inv}}} d \qquad \int_{Z^e_{\text{inv}} \setminus Z^e_{\text{inv}}} g_s d \qquad \frac{1}{s} (P_s) > 0 \tag{68}$$

$(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) > 0 \implies \mathsf{P}(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) > 0 \implies \mathsf{P}(Z^e_{\text{inv}}) < 1$ which is a contradiction. Therefore, $(Z^e_{\text{inv}} \cap Z^e_{\text{inv}}) = 0$.

We now describe how our assumptions already eliminate the possibility of distribution shifts that happen in such a way that the a finite mass of the distribution resides in the region $Z^e_{\text{inv}} \cap \tilde{Z}^e_{\text{inv}}$. Recall we assume that $\forall e \in E_{all}$, Z^e_{inv} is a continuous random variable. Since the probability of continuous random is absolutely continuous w.r.t the Lebesgue measure it follows that for each $e \in E_{all}$, $(Z^e_{\text{inv}} \cap \tilde{Z}^e_{\text{inv}}) = 0 \implies P(Z^e_{\text{inv}} \cap \tilde{Z}^e_{\text{inv}}) = 0$. Thus all distribution shifts would place a zero mass in the region of disagreement.

This completes the first part of the proof.

The second part of the theorem follows directly from the analysis of linear regression SEM in Arjovsky et al. (2019). The conditions in the second part of the theorem cover the conditions that are required in Theorem 1. Under those conditions there can be two invariant predictors one is the trivial invariant predictor that maps every input to zero. The other is the ideal invariant predictor that focuses on the causes. The constraint r^{th} is set to a low enough value such that only the ideal invariant predictor gets selected. Observe that the risk achieved by the trivial zero invariant predictor is $\frac{1}{|\mathcal{E}_{tr}|} E[(Y^e)^2] = \sigma^2_Y$ and the risk achieved by the ideal $\frac{1}{|\mathcal{E}_{tr}|} E[(N^e)^2] = \sigma^2_N$. If $\sigma^2_N < r^{\text{th}} < \sigma^2_Y$, then the only predictor that is selected is the ideal invariant predictor.

We now describe why ERM fails in this case. In the theorem, we assume that $\exists\, e$ where $v = \mathrm{E}[\epsilon^e Z^e_{\text{spu}}] \neq 0$, which implies $\mathrm{E}[\epsilon^e X^e] \neq 0$. We show why this is the case next.

$$\mathrm{E}[\epsilon^e X^e] = \mathrm{E}[\epsilon^e S(Z^e_{\text{inv}}, Z^e_{\text{spu}})] = \mathrm{E}[S^{\epsilon^e}(Z^e_{\text{inv}}, Z^e_{\text{spu}})] = S(0, v) \neq 0, \text{since } S \text{ is invertible} \tag{69}$$

The rest of the proof follows from Proposition 17 in (Ahuja et al., 2021b). If r^{th} is set low enough to assume the same risk achieved by ERM, then IB-ERM and ERM are identical and IB-ERM also fails.

□

Remark on invertibility of S. The entire proof extends to the case when S is not invertible but Z^e_{inv} can still be recovered. Note that at no point in the proof we required to have full S to be invertible.

Remark on regularized ERM, IRM. Note that while we showed that the ERM and IRM fail, the failures extend to ℓ_1 or ℓ_2 regularized models as well. We would like to also mention that it may seem that information bottleneck and sparsity constraints such as ℓ_1 have similarity. We want to point out that there is a major difference between the two. In our model, we observe scrambled data. As a result, even if there is sparsity in the latent space, that does not translate to the observed space. ℓ_1 constraints operate in the input space and that is why they cannot fetch the same outcome as information bottleneck constraints.

Remark on multi-class classification. The proof presented in this section extends to multi-class setting described in Assumption 10. The simplification in equation (53) along with the lemmas (Lemma 6, Lemma 7) help establish why low-entropy representation based classifier discourages the use of spurious features. We can adapt the analysis in equation (53) to the multi-class case (Assumption 10) and follow the same line of reasoning to justify why IB-IRM and IB-ERM succeed.

A.7 Derivation of the final objective in equation (7)

In this section, we give a step-by-step description of derivation of the objective in equation (7). We rewrite the IB-IRM optimization below in equation (70).

$$\begin{aligned} &\min_{\in \mathrm{R}^k} \frac{1}{jE_{tr}j} \sum_e h^e \; w \\ &\text{s.t.} \; \frac{1}{jE_{tr}j} \sum_e R^e \; w \qquad r^{\text{th}}; \\ &\quad 1 \; 2 \; \arg\min_{w \in \mathrm{R}} R^e(w \quad): \end{aligned} \tag{70}$$

In the above we assumed that the classifiers are scalar. We state a new optimization that we show is equivalent to the optimization in equation (70).

$$\begin{aligned} &\min_{\in \mathrm{R}^k} \frac{1}{jE_{tr}j} \sum_e h^e \\ &\text{s.t.} \; \frac{1}{jE_{tr}j} \sum_e R^e \qquad r^{\text{th}}; \\ &\quad 1 \; 2 \; \arg\min_{w \in \mathrm{R}} R^e(w \quad): \end{aligned} \tag{71}$$

It can be shown that the two forms of optimization in equation (70) and equation (71) are equivalent. First, we would like to show that the set of feasible classifiers w for the first optimization in equation (71) and in the second optimization in equation (71) are the same.

Suppose $w^*; \; ^*$ is a feasible solution to the constraints in equation (70). Construct $^\dagger = w^* \; ^*$. † satisfies the constraint $\frac{1}{|\mathcal{E}_{tr}|} \sum_e R^e \; ^\dagger \qquad r^{\text{th}}$. Suppose for some environment e, $1 \; 2 \; \arg\min_w R^e(w \quad ^\dagger) \; =) \; \; 9 \; w \; 6 = 1$ such that $R^e(w \quad ^\dagger) < R^e(\; ^\dagger)$. If this is the case, then $w \quad w^*$ improves over w^* and contradicts the optimality of w^* in equation (70). This establishes that † satisfies the constraints in equation (70). This shows that the set of feasible classifiers for the first optimization in equation (70) are a subset of the feasible classifiers in the second optimization (71).

Suppose * is a feasible solution to the constraints in equation (71). Take any scalar w and corresponding representation $^*=w$. The combined classifier $w \quad (\; ^*=w)$ satisfies the first constraint. Suppose $w \; 2 \; \arg\min_{w \in \mathrm{R}} R^e(w \; \frac{}{w})$, this implies that $9 \; w^+ \; 6 = w$ such that $R^e(\frac{w^+}{w} \qquad ^*) < R^e(\; ^*)$. If this was true, then that contradicts the optimality of 1 in equation (71). This shows that the set of feasible classifiers for the second optimization in equation (71) are a subset of the feasible classifiers in the first optimization (70).

From the above discussion, it is clear that the two formulations result in the same set of feasible w , which are finally fed into the same entropy minimization objective. Thus the two optimizations are equivalent. To get to the penalized objective in equation (7) from the equation (71) there are two key steps: i) converting the invariance constraint into the gradient-based penalty, i.e., the IRMv1 penalty from (Arjovsky et al., 2019), ii) converting the differential entropy term into a constraint on the variance. For ii), as we explained in the manuscript, minimization of variance is equivalent to minimizing an upper bound on the entropy. Also, note that since variance has a lower bound, we can directly work with and do not need to add a noise term like earlier, which was done to ensure that differential entropy is lower bounded. Below we break down the steps to arrive at the objective. We first start with a weighted combination of the terms in equation (6).

$$\sum_e R^e(\;) + \; k r_{w;w=1.0} R^e(w \quad)k^2 + \; h^e(\;) \; : \tag{72}$$

where $k r_{w;w=1.0} R^e(w \quad)k^2$ is the norm of the gradient computed w.r.t scalar classifier w at 1.0. Note that in general the gradient can be computed w.r.t a fixed vector as well. In our experiments, we

found that using entropy conditioned on the environment or entropy unconditioned on the environment works equally well. Thus, we introduce the unconditional entropy $h(\ )$. We assume that all the environments occur with an equal probability.

$$h(\ ) = \mathsf{E}_{X\sim\mathsf{P}}[\log(d\mathsf{P}(\ (X))] \tag{73}$$

where $d\mathsf{P}(\ (X))$ is the probability density of predictions (unconditional on the environment), $\mathsf{P} = \frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}} \mathsf{P}^e$ is the uniform mixture of data from all environments. Note here X denotes an input sample and we do not know the environment it comes from unlike the sample X^e. The entropy of predictions computed in environment e is given as

$$h^e(\ ) = \mathsf{E}_{X^e\sim\mathsf{P}^e}[\log(d\mathsf{P}^e(\ (X^e))]; \tag{74}$$

where $d\mathsf{P}^e$ is the probability density of the predictions in environment e. The conditional entropy over predictions conditioned on a random environment is given as

$$h(\ |E) = \frac{1}{|E_{tr}|}\sum_{e\in\mathcal{E}_{tr}} \mathsf{E}[\log(d\mathsf{P}^e(\ (X^e))]: \tag{75}$$

Conditioning reduces entropy $h(\ ) \geq h(\ |E)$ and thus we propose an upper bound on the objective in equation (72) below

$$\sum_e R^e(\ ) + \ k\ r_{w;w=1:0}R^e(w\ \ )k^2 + \ h(\ )\ : \tag{76}$$

Finally, instead of $h(\ )$ we use variance in predictions denoted as $\mathsf{Var}(\ ) = \mathsf{E}_{X\sim\mathsf{P}}[(\ (X) - \mathsf{E}[\ (X)])^2]$ to get

$$\sum_e R^e(\ ) + \ k\ r_{w;w=1:0}R^e(w\ \ )k^2 + \ \mathsf{Var}(\ )\ : \tag{77}$$

A.8 Proof of Theorem 5: impact of IB on the learning speed

In this section, we present a detailed analysis of 2D case in equation (4) leading up to the proof of Theorem 5. For convenience, we will restate the equation (4). Also, instead of assuming the binary values are from the set $\{0, 1\}$ we would shift them to $\{-1, 1\}$; we do this purely for making notation clearer.

$$\begin{aligned} Y^e &= \text{sgn}\left(X^e_{\text{inv}}\right), \text{ where } X^e_{\text{inv}} = -1 \text{ with probability } \frac{1}{2}, \\ X^e_{\text{spu}} &= X^e_{\text{inv}} W^e, \text{ where } W^e = -1 \text{ with probability } 1 - p^e \text{ with selection bias } p^e > \frac{1}{2}, \end{aligned} \tag{78}$$

Connection between the discrete and the continuous case. Before discussing the proof of Theorem 5, we provide an explanation as to why can we use the variance penalty as a proxy for the 2D example (eq. (78)), where the random variables are discrete (recall that variance is monotonically related to upper bound on the differential entropy of continuous random variables). We present a variation of equation (78), where the input feature values are continuous. For each $e \in \mathcal{E}_{tr}$ we have

$$\begin{aligned} X^e_{\text{inv}} &= C^e + U^e, \\ Y^e &= \text{sgn}(X^e_{\text{inv}}), \end{aligned} \tag{79}$$

where $C^e = -1$ with equal probability for -1 and 1 and U^e is a uniform random variable with range $[-r, r]$ with $r < \frac{1}{2}$. Similarly, with probability $1 - p^e$,

$$X^e_{\text{spu}} = C^e + M^e,$$

and with probability p^e,

$$X^e_{\text{spu}} = -C^e + M^e,$$

where M^e is a uniform random variable with range $[-r, r]$.

Suppose ℓ is exponential loss and the predictor has two dimensions w_{inv} and w_{spu}. For the above problem description, we write the ERM objective ($\alpha = 0$, $\gamma = 0$ in equation (7)) and we get the following

$$\begin{aligned} &R_{\text{ERM}}(w_{\text{inv}}, w_{\text{spu}}) = \\ &\frac{1}{|\mathcal{E}_{tr}|} \sum_{e \in \mathcal{E}_{tr}} p^e e^{-(w_{\text{inv}} + w_{\text{spu}})} \mathbb{E}[e^{-w_{\text{inv}} U^e} e^{-w_{\text{spu}} M^e}] + (1 - p^e) e^{-(w_{\text{inv}} - w_{\text{spu}})} \mathbb{E}[e^{-w_{\text{inv}} U^e} e^{w_{\text{spu}} M^e}] \\ &\mathbb{E}[e^{-w_{\text{inv}} U^e} e^{-w_{\text{spu}} M^e}] = \mathbb{E}[e^{-w_{\text{inv}} U^e}] \mathbb{E}[e^{-w_{\text{spu}} M^e}] \\ &\mathbb{E}[e^{-w_{\text{inv}} U^e}] = \int_{-r}^{r} e^{-w_{\text{inv}} u} du \; \frac{1}{2r} = \frac{e^{w_{\text{inv}} r} - e^{-w_{\text{inv}} r}}{2 w_{\text{inv}} r} \approx \frac{(1 + w_{\text{inv}} r) - (1 - w_{\text{inv}} r)}{2 w_{\text{inv}} r} = 1 \end{aligned} \tag{80}$$

If r is small, then we can approximate the loss as if the each of the feature values were discrete and only assumed one of the four possible values in $\{-1, 1\} \times \{-1, 1\}$.

$$R_{\text{ERM}}(w_{\text{inv}}, w_{\text{spu}}) \approx p e^{-(w_{\text{inv}} + w_{\text{spu}})} + (1 - p) e^{-(w_{\text{inv}} - w_{\text{spu}})} \tag{81}$$

where $p = \frac{1}{|\mathcal{E}_{tr}|} p^e$. On the same lines, we expand the IB-ERM objective as follows

$$R_{\text{IB-ERM}}(w_{\text{inv}}, w_{\text{spu}}) \approx p e^{-(w_{\text{inv}} + w_{\text{spu}})} + (1 - p) e^{-(w_{\text{inv}} - w_{\text{spu}})} + [w_{\text{inv}}, w_{\text{spu}}] \, \Gamma \, [w_{\text{inv}}, w_{\text{spu}}]^\top \tag{82}$$

where $\Gamma = \begin{bmatrix} 1 + r^2 & 2p - 1 \\ 2p - 1 & 1 + r^2 \end{bmatrix}$. Since r is small, we approximate Γ as $\begin{bmatrix} 1 & 2p - 1 \\ 2p - 1 & 1 \end{bmatrix}$.

Theorem on impact of information bottleneck. We would compare the rate of convergence of continuous-time gradient descent for $R_{\text{IB-ERM}}$ and R_{ERM}.

Theorem 10. *Suppose each $e \in E_{tr}$ follows the 2D case from equation* (4)*. Set* $= 0$, > 0 *in equation* (7) *to get the IB-ERM objective with* ` *as exponential loss. Continuous-time gradient descent on this IB-ERM objective achieves* $j\frac{w_{spu}(t)}{w_{inv}(t)}j$ *in time less than* $\frac{W_0(\frac{1}{2})}{2(1-p)}$ *($W_0(\ )$ denotes the principal branch of the Lambert W function), while in the same time the ratio for ERM* $j\frac{w_{spu}(t)}{w_{inv}(t)}j$ $\ln(\frac{1+2p}{3-2p})=\ln\ 1+\frac{W_0(\frac{1}{2})}{2(1-p)}$ *, where* $p = \frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}} p^e$.

Proof of Theorem 10. We simplify the ERM and the IB-ERM objective in equation (7) for the 2D case.

$$R_{\text{ERM}}(w_{\text{inv}}; w_{\text{spu}}) = pe^{-(w_{\text{inv}}+w_{\text{spu}})} + (1 \quad p)e^{-(w_{\text{inv}}-w_{\text{spu}})}$$

$$R_{\text{IB-ERM}}(w_{\text{inv}}; w_{\text{spu}}) = pe^{-(w_{\text{inv}}+w_{\text{spu}})} + (1 \quad p)e^{-(w_{\text{inv}}-w_{\text{spu}})} + \ [w_{\text{inv}}; w_{\text{spu}}]\ \ [w_{\text{inv}}; w_{\text{spu}}]^{\mathsf{T}}$$

where $w_{\text{inv}}; w_{\text{spu}} \in \mathbb{R}$ are the weights for invariant and spurious features, $p = \frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}} p^e$ as $\begin{matrix} 1 & 2p \\ 2p & 1 \end{matrix}\ \begin{matrix} 1 \\ 1 \end{matrix}$. We first find the equilibrium point of the continuous-time gradient descent for $R_{\text{IB-ERM}}$.

$$\begin{aligned}
\frac{\partial R_{\text{IB-ERM}}(w_{\text{inv}}; w_{\text{spu}})}{\partial w_{\text{inv}}} &= \quad pe^{-(w_{\text{inv}}+w_{\text{spu}})} \quad (1 \quad p)e^{-(w_{\text{inv}}-w_{\text{spu}})} + 2\ (w_{\text{inv}} + (2p \quad 1)w_{\text{spu}}) \\
\frac{\partial R_{\text{IB-ERM}}(w_{\text{inv}}; w_{\text{spu}})}{\partial w_{\text{spu}}} &= \quad pe^{-(w_{\text{inv}}+w_{\text{spu}})} + (1 \quad p)e^{-(w_{\text{inv}}-w_{\text{spu}})} + 2\ ((2p \quad 1)w_{\text{inv}} + w_{\text{spu}})
\end{aligned} \tag{83}$$

$$\begin{aligned}
&\frac{\partial R_{\text{IB-ERM}}(w_{\text{inv}}; w_{\text{spu}})}{\partial w_{\text{inv}}} + \frac{\partial R_{\text{IB-ERM}}(w_{\text{inv}}; w_{\text{spu}})}{\partial w_{\text{spu}}} = \quad 2pe^{-(w_{\text{inv}}+w_{\text{spu}})} + 4\ p(w_{\text{inv}} + w_{\text{spu}}) = 0 \\
&=) \quad \frac{1}{2}e^{-(w_{\text{inv}}+w_{\text{spu}})} = w_{\text{inv}} + w_{\text{spu}} \\
&=) \quad w_{\text{inv}} + w_{\text{spu}} = W_0\ \frac{1}{2}
\end{aligned} \tag{84}$$

$$\begin{aligned}
&\frac{\partial R_{\text{IB-ERM}}(w_{\text{inv}}; w_{\text{spu}})}{\partial w_{\text{inv}}} \quad \frac{\partial R_{\text{IB-ERM}}(w_{\text{inv}}; w_{\text{spu}})}{\partial w_{\text{spu}}} = \quad 2(1 \quad p)pe^{-(w_{\text{inv}}-w_{\text{spu}})} + 4\ (1 \quad p)(w_{\text{inv}} \quad w_{\text{spu}}) = 0 \\
&=) \quad \frac{1}{2}e^{-(w_{\text{inv}}-w_{\text{spu}})} = w_{\text{inv}} \quad w_{\text{spu}} \\
&=) \quad w_{\text{inv}} \quad w_{\text{spu}} = W_0\ \frac{1}{2}
\end{aligned} \tag{85}$$

Therefore, the equilibrium point is $w_{\text{inv}} = W_0\ \frac{1}{2}$ and $w_{\text{spu}} = 0$. Having established that the equilibrium point of the differential equation coincides with ideal predictor, we now analyze the convergence of the trajectory. Let $w_{\text{inv}} + w_{\text{spu}} = x$ and $w_{\text{inv}} \quad w_{\text{spu}} = y$.

$$\frac{\partial x}{\partial t} = \quad \frac{\partial R_{\text{IB-ERM}}(w_{\text{inv}}; w_{\text{spu}})}{\partial w_{\text{inv}}} + \frac{\partial R_{\text{IB-ERM}}(w_{\text{inv}}; w_{\text{spu}})}{\partial w_{\text{spu}}} = 2p(e^{-x} \quad 2\ x) \tag{86}$$

$$\frac{\partial y}{\partial t} = 2(1 \quad p)(e^{-y} \quad 2\ y) \tag{87}$$

Let us call $x^* = W_0\ \frac{1}{2}$; x^* is equilibrium point for both $x(t)$ and $y(t)$. Denote $w_{\text{inv}}(t) = \frac{x(t)+y(t)}{2}$ and $w_{\text{spu}}(t) = \frac{x(t)-sy(t)}{2}$. Let us assume that $x(0) = 0$ and $y(0) = 0$. We would first like to argue that the solution to the above differential equations exist and are unique given the initial conditions

$x(0) = 0$ and $y(0) = 0$. Since $(e^{-x} \quad 2 \ x)$ is Lipschitz continuous in x on $\mathbb{R}$ the solution to the differential equation exists and is unique for any finite interval $t \ 2 \ [0; T]$ (Simmons, 2016). With T set to a sufficiently large value, we now show that the solution to the ODE converges to x^*.

Define an energy function $V(z) = z^2$ and define $V(x \quad x^*) = (x \quad x^*)^2$

$$\frac{@V(x \quad x^*)}{@t} = 2(x \quad x^*)\frac{@x}{@t} = 4p(x \quad x^*)(e^{-x} \quad 2 \ x) \tag{88}$$

Observe that $\frac{@V(x-x\ )}{@t} < 0$ for all $x \neq x^*$ and $\frac{@V(x-x\ )}{@t} = 0$ when $x = x^*$. Therefore, from Lyapunov's asymptotic global stability theorem (Khalil, 2009) we obtain that $x(t)$ would converge to x^*.

Observe that for $x < x^*$, $\frac{@x}{@t} > 0$ and moreover $2p(e^{-x} \quad 2 \ x)$ is a monotonically decreasing function. For all $x < x^* \quad$, we can bound the rate at which x increases is bounded below by $2p(e^{-x\ +} \quad 2 \ (x^* \quad)) \quad 2p(e^{-x} \ (1 + \ ) \quad 2 \ x^* + 2 \ ) = 2p \ (e^{-x} \quad + 2 \ )$. Let us call $^* = (e^{-x} \quad + 2 \ )$. The rate at which x increases is greater than $2p \quad ^*$ and the rate at which y increases is greater than $2(1 \quad p) \quad ^*$. Thus the time to convergence for x is atmost $\frac{x}{2p}$. Similarly, the time to convergence for y is atmost $\frac{x}{2(1-p)}$. Since $p > \frac{1}{2}$ the time to convergence for $y(t)$ is more than the time taken for the convergence of $x(t)$.

If $jx(t) \quad x^*j \quad$ and $jy(t) \quad x^*j \quad$, then $jw_{\text{spu}}(t)j = j\frac{x(t)-y(t)}{2}j = j\frac{x(t)-x\ +x\ -y(t)}{2}j$ $\frac{|x(t)-x\ |}{2} + \frac{|y(t)-x\ |}{2}$.

If $jx(t) \quad x^*j \quad$ and $jy(t) \quad x^*j \quad$, then $jw_{\text{inv}}(t) \quad x^*j = j\frac{x(t)+y(t)}{2} \quad x^*j = j\frac{x(t)-x\ +y(t)-x}{2}j$ $\frac{|x(t)-x\ |}{2} + \frac{|y(t)-x\ |}{2}$.

As a result, if $jx(t) \quad x^*j \quad$ and $jy(t) \quad x^*j \quad$, then

$$\frac{jw_{\text{spu}}(t)j}{jw_{\text{inv}}(t)j} \quad \frac{}{x^*} \quad \overline{x^*} \tag{89}$$

Therefore, to get the ratio $\frac{|w_{\text{spu}}(t)|}{|w_{\text{inv}}(t)|} \quad \overline{x}$ the time taken is at most $\frac{x}{2(1-p)}$.

In comparison in the same amount of time the ratio $j\frac{w_{\text{spu}}(t)}{w_{\text{inv}}(t)}j$ achieved by gradient descent on R_{ERM} is at least $\frac{\ln(\frac{1+2p}{3\ 2p})}{\ln(1+\frac{x}{2(1\ \ p)})}$. The expression for lower bound on the ratio $j\frac{w_{\text{spu}}(t)}{w_{\text{inv}(t)}}j$ is derived by substituting the time taken, i.e., $\frac{x}{2(1-p)}$, in the expression for the lower bound derived in Section B.3 in Nagarajan et al. (2021)). □

Remark on max-margin classifiers. In the 2D example, the max-margin classifier seems to solve the problem. In general max-margin classifier would not work. In the more general setting, if there is noise in the labels, which is allowed by the SEM in Assumption 8, and the data is scrambled, which is also the case in Assumption 8, there is no guarantee that max-margin classifier would not rely on the spurious features.

A.9 Illustrating both invariance and information bottleneck acting in conjunction

In this section, we present a case to illustrate why the invariance principle and the information bottleneck are needed simultaneously. The model we present follows a DAG that combines the DAGs in Figure 2a) and Figure 2b).

Example extending the 2D case from equation (4)**.** For all the environments $e \in E_{tr}$

$$\begin{array}{ccc} Y^e & X^e_{\text{inv}} & N^e \\ X^{1;e}_{\text{spu}} & Y^e & W^e \\ X^{2;e}_{\text{spu}} & X^e_{\text{inv}} & V^e \end{array} \tag{90}$$

where all the variables in the above SEM are binary $\{0, 1\}$ random variables. $N^e \sim \text{Bernoulli}(q)$, $V^e \sim \text{Bernoulli}(a)$; the distribution of noise N^e and V^e are the same across the environments. $W^e \sim \text{Bernoulli}(u^e)$ where u^e is an environment dependent probability. For all the environments $e \in E_{all}$, we assume that the distribution of X^e_{inv}, N^e, and V^e does not change. The labelling function to generate Y^e is also the same. The distribution of $X^{1;e}_{\text{spu}}$ can change arbitrarily. In this example, observe that $\mathrm{E}[Y^e|X^e]$ varies across the training environments. We show the simplification below.

$$\mathrm{E}[Y^e|X^e] = \mathrm{E}\left[X^e_{\text{inv}} \oplus N^e \,(X^e_{\text{inv}}, X^{1;e}_{\text{spu}}, X^{2;e}_{\text{spu}})\right] \tag{91}$$

If $X^e_{\text{inv}} = 0, X^e_{\text{spu}} = 0$, then $\mathrm{E}[Y^e|X^e] = \mathrm{P}(N^e = 1|X^e_{\text{inv}} = 0, X^{1;e}_{\text{spu}} = 0)$. We show that $\mathrm{P}(N^e = 1|X^e_{\text{inv}} = 0, X^{1;e}_{\text{spu}} = 0)$ varies across the environments.

$$\begin{aligned} \mathrm{P}(N^e = 1|X^e_{\text{inv}} = 0, X^{1;e}_{\text{spu}} = 0) &= \frac{\mathrm{P}(N^e = 1, X^e_{\text{inv}} = 0, X^{1;e}_{\text{spu}} = 0)}{\mathrm{P}(N^e = 1, X^e_{\text{inv}} = 0, X^{1;e}_{\text{spu}} = 0) + \mathrm{P}(N^e = 0, X^e_{\text{inv}} = 0, X^{1;e}_{\text{spu}} = 0)} \\ &= \frac{\mathrm{P}(N^e = 1, X^e_{\text{inv}} = 0)u^e}{\mathrm{P}(N^e = 1, X^e_{\text{inv}} = 0)u^e + \mathrm{P}(N^e = 0, X^e_{\text{inv}} = 0)(1 - u^e)} \end{aligned} \tag{92}$$

Note that the above equation (92) describes the probability computed by the Bayes optimal classifier that relies on input feature dimensions are used. Observe that the above probability in equation (92) can only be equal across two environments if u^e was the same. Therefore, if $|E_{tr}| \geq 2$ and the probability u^e varies across the environments, then the invariance constraint restrict us from using the identity representation. However, $\mathrm{E}[Y^e|X^e_{\text{inv}}, X^{2;e}_{\text{spu}}]$ is invariant and so is $\mathrm{E}[Y^e|X^e_{\text{inv}}]$. Based on the same arguments that we discussed in the main manuscript, we can show that one can construct classifiers that output probability distributions that minimize cross-entropy (maximize likelihood) and continue to depend on $X^{2;e}_{\text{spu}}$ as follows

$$\begin{aligned} \hat{\mathrm{P}}(Y^e = 1|X^e_{\text{inv}}, X^{2;e}_{\text{spu}}) = (1 - q)\mathbb{I}\left[w_{\text{inv}}X^e_{\text{inv}} + w_{\text{spu}}X^e_{\text{spu}} > \frac{(w_{\text{inv}} + w_{\text{spu}})}{2}\right] + \\ q\,\mathbb{I}\left[w_{\text{inv}}X^e_{\text{inv}} + w_{\text{spu}}X^e_{\text{spu}} < \frac{(w_{\text{inv}} + w_{\text{spu}})}{2}\right]. \end{aligned} \tag{93}$$

If $w_{\text{inv}} > |w_{\text{spu}}|$, then above classifier $\hat{\mathrm{P}}(Y^e = 1|X^e_{\text{inv}}, X^{2;e}_{\text{spu}})$ matches the true probability distribution conditional on the invariant feature $\mathrm{P}(Y^e = 1|X^e_{\text{inv}})$ on all the training environments and it thus forms a valid invariant predictor with representation that focuses on $X^e_{\text{inv}}, X^{2;e}_{\text{spu}}$. Since the classifier relies on $X^{2;e}_{\text{spu}}$, the classifier fails as the support of spurious features can change. If we place an entropy constraint, then the representation that focuses only on X^e_{inv} is strictly prefered to one that focuses on both $X^e_{\text{inv}}, X^{2;e}_{\text{spu}}$ and continues to achieve the same cross-entropy loss. Thus in this example, IRM fails as its solution space contains classifiers that rely on spurious features but IB-IRM would succeed. In the above example, ERM and IB-ERM (with r^{th} set to match the loss of ERM) will rely on $X^{1;e}_{\text{spu}}$ on top of X^e_{inv} as conditioning on $X^{1;e}_{\text{spu}}$ in addition to X^e_{inv} further reduces the conditional entropy thus reducing the cross-entropy loss.

Let us consider a generalization of the above example.

Assumption 11. *Each environment $e \in E_{all}$ follows*

$$Y^e = \mathrm{I}\, w^*_{\text{inv}} \quad X^e_{\text{inv}} \qquad N^e \tag{94}$$

N^e is binary noise, and X^e_{inv} are binary features. Both N^e and X^e_{inv} have identical distributions across all the environments E_{all}

Divide the spurious features into two parts $X^e_{\text{spu}} = (X^{1;e}_{\text{spu}}; X^{2;e}_{\text{spu}})$.

Assumption 12. *Each environment $e \in E_{tr}$ follows*

$$\begin{aligned} X^{1;e}_{\text{spu}} \qquad & Y^e 1 \quad W^e \\ X^{2;e}_{\text{spu}} \qquad & X^e_{\text{inv}} \quad V^e \end{aligned} \tag{95}$$

where $1 \in \mathrm{R}^{o'}$ is a vector of ones, $W^e \in \mathrm{R}^{o'}$ is a binary 0-1 vector with each component drawn i.i.d. from $\text{Bernoulli}(u^e)$ vector, V^e is also a binary 0-1 vector with each component drawn i.i.d. from $\text{Bernoulli}(a)$ vector. The distribution of W^e changes across environments and no two training environments have the same u^e. The distribution of V^e is identical across all the training environments. Also, assume that there are at least two training environments, i.e., $|E_{tr}| \geq 2$.

Assumption 13. *H is a set of diagonal matrices, where each element in the matrix is 0 or 1 (H act as matrices that seletct subset of input features). H_w is set of all probability distributions on R^d. ℓ is the cross-entropy loss.*

We use the Shannon entropy formulation of IB-IRM in this case as all the random variables involved are discrete. Moreover, we carry out entropy minimization for the representation directly and not the predictor. The IB-IRM optimization is given as follows.

$$\begin{aligned} \min_{\in\mathcal{H}} \quad & \frac{1}{|E_{tr}|} \sum_{e} H^e(\ ) \\ \text{s.t.} \quad & \frac{1}{|E_{tr}|} \sum_{e} R^e \ w \qquad\qquad r^{\text{th}} \\ & w \in \arg\min_{w\in\mathcal{H}_w} R^e(w \qquad) \end{aligned} \tag{96}$$

Theorem 11. *Suppose the data follows Assumption 11, Assumption 12. Suppose H_w and H follow Assumption 13. If invariant features are strictly separable, i.e., Assumption 7 holds, then IRM fails but IB-IRM succeeds.*

Proof of Theorem 11. We carry out the analysis for different types of representations separately.

Case 1: Consider a representation that selects a subset $\tilde{X}^e_1$ of $(X^e_{\text{inv}}; X^{2;e}_{\text{spu}})$ and a subset $\tilde{X}^e_2$ of $X^{1;e}_{\text{spu}}$.

$$\begin{aligned} \mathrm{P}(Y^e = 1 | \tilde{X}^e_1 = 0; \tilde{X}^e_2 = 0) &= \frac{\mathrm{P}(Y^e = 1; \tilde{X}^e_1 = 0; \tilde{X}^e_2 = 0)}{\mathrm{P}(Y^e = 1; \tilde{X}^e_1 = 0; \tilde{X}^e_2 = 0) + \mathrm{P}(Y^e = 0; \tilde{X}^e_1 = 0; \tilde{X}^e_2 = 0)} \\ &= \frac{\mathrm{P}(Y^e = 1; \tilde{X}^e_1 = 0)(u^e)^{o'}}{\mathrm{P}(Y^e = 1; \tilde{X}^e_1 = 0)(u^e)^{o'} + \mathrm{P}(Y^e = 1; \tilde{X}^e_1 = 0)(1 \quad u^e)^{o'}} \end{aligned} \tag{97}$$

Since $\mathrm{P}(Y^e = 1 | \tilde{X}^e_1 = 0; \tilde{X}^e_2 = 0)$ is strictly monotonic in u^e, this probability cannot be same across two environments. Hence, any $\tilde{X}^e_1; \tilde{X}^e_2$ cannot lead to an invariant predictor across the two environments.

Case 2: Consider a representation that selects a subset $\tilde{X}^e$ of $X^{1;e}_{\text{spu}}$.

$$\begin{aligned} \mathrm{P}(Y^e = 1 | \tilde{X}^e = 0) &= \frac{\mathrm{P}(Y^e = 1; \tilde{X}^{1;e} = 0)}{\mathrm{P}(Y^e = 1; \tilde{X}^{1;e} = 0) + \mathrm{P}(Y^e = 0; \tilde{X}^{1;e} = 0)} \\ &= \frac{\mathrm{P}(Y^e = 1)(u^e)^{o'}}{\mathrm{P}(Y^e = 1)(u^e)^{o'} + \mathrm{P}(Y^e = 0)(1 \quad u^e)^{o'}} \end{aligned} \tag{98}$$

For the above class of representations also, we can use the same argument as the one discussed in Case 1 and show that the above probability cannot be the same across two environments.

Case 3: At this point, our only option is to consider representations that select subsets of $(X^e_{\text{inv}}; X^{2;e}_{\text{spu}})$. Each subset of $(X^e_{\text{inv}}; X^{2;e}_{\text{spu}})$ satisfies invariance. Among this set all the subsets that lead to lowest cross-entropy are selected by IRM. Among those sets IRM does not exclude the inclusion of spurious covariates $X^{2;e}_{\text{spu}}$. However, when we impose entropy minimization objective, then $X^{2;e}_{\text{spu}}$ will never be selected as entropy can be strictly reduced by not including these covariates in the set without sacrificing invariance or cross-entropy. To explicitly show a construction of the failure of IRM in this case, we can use the same construction as equation (93) but replacing the hyperplane in the indicator function with hyperplane constructed in Lemma 4.

□

A.10 Related works

A.10.1 Invariance principles in causality

The foundations of invariance principles are rooted in the theory of causality (Pearl, 1995). There are several different forms in which the invariance principles or principles similar to it appear in the literature on causality. Modularity condition states that a variable Y is caused by a set of variables $X_{\mathrm{Pa}(Y)}$if and only if under all interventions other than those on Y the conditional probability $P(Y|X_{\mathrm{Pa}(Y)})$ remains invariant. Related and similar notions are *stability* (Pearl, 2009), *autonomy* (Schölkopf et al., 2012), *invariant causal prediction principle* (Peters et al., 2016; Heinze-Deml et al., 2018). These principles lead to a powerful insight – if we model all the environments (train and test) using interventions, then as long as these interventions do not affect the causal mechanism that generates the target variable Y, a classifier trained only on the transformation that extracts causal variables ($(X) = X_{\mathrm{Pa}(y)}$) to predict Y is invariant under interventions.

A.10.2 Invariance principles in OOD generalization

In recent years, there has been a surge in the works inspired from causality, examples of some notable works are (Peters et al., 2016; Arjovsky et al., 2019), which seek to address OOD generalization failures. The invariance principle is at the heart of many of these works. For a better understanding, we divide these works into two categories – theory and methods, though some works belong to both.

Theory. In Rojas-Carulla et al. (2018) it was shown that the predictors trained on the causes are min-max optimal under a large class of distribution shifts modeled by the interventions. These findings were generalized in Koyama and Yamaguchi (2020). Given that we know that predictors that focus on the causes are min-max optimal under many distribution shifts, the central question then is – can we learn these predictors from a finite set of training distributions/environments? Arjovsky et al. (2019) showed how to achieve such causal predictors that generalize OOD from a finite set of training environments for linear regression tasks under very general assumptions. Rosenfeld et al. (2021b) considered linear classification tasks where invariant features were partially informative w.r.t the label and showed that under assumptions of support overlap for invariant and spurious features, it is possible to learn predictors that generalize OOD. In this work, we analyze classification tasks but different from Rosenfeld et al. (2021b) we consider both fully and partially informative features. We showed that support overlap of invariant features is necessarily needed for OOD generalization in classification tasks else OOD generalization, in general, is impossible. On the other hand, we showed that support overlap for spurious features is not necessary but in its absence standard methods such as ERM and IRM can fail.

Recent works (Rosenfeld et al., 2021b,a; Kamath et al., 2021; Gulrajani and Lopez-Paz, 2021) have also pointed to several limitations of invariance based approaches for addressing OOD generalization failures. In Rosenfeld et al. (2021b), the authors showed that if we use the IRMv1 objective, then for non-linear tasks the solutions from IRMv1 are no better than ERM in generalizing OOD. In Lu et al. (2021), the authors present a two-phased approach to addressing the difficulties faced by IRM in the non-linear regime. In the first phase, an identifiable variational autoencoder (Khemakhem et al., 2020) is used to extract the latent representations from the raw input data. In the second phase, causal discovery-based approaches are used to identify the causal parents of the label and then learn predictors based on the causal parents only. The entire analysis in Lu et al. (2021) is for the setting when the invariant features are partially informative about the label. Also, the analysis assumes that we have access to side information (possibly in the form of environment index) that can help disentangle all the latent features, i.e., all the latent features are independent conditioned on this side information. Having access to such information, in general, is a strong assumption. In Kamath et al. (2021), the authors show that if the label and feature space is finite and if the distribution shifts are captured by analytic functions, then the set of invariant predictors found from two environments exactly capture all the invariant predictors described by the analytic function. While this is a very interesting and important result, we would like to point out that the distribution shifts captured using analytic functions represent a small family of interventions that are otherwise allowed when learning predictors that focus on causes.

In this work, we focused on linear SEMs unlike the non-linear SEMs described above. The setting that we considered in this work has three salient features – a) classification when invariant features are fully informative, b) spurious features are correlated with invariant features, and c) arbitrary shifts

are allowed on the spurious feature distribution. This setting is important as many of the existing failures correspond to this setting. We are the first to give provable OOD generalization guarantees for this setting. Considering non-linear models is a natural next step. On this note, we would like to mention that we believe several of our results can be generalized to the setting when the mapping from the latents to the raw data is piecewise linear.

Methods. Following the original works ICP (Peters et al., 2016) and IRM (Arjovsky et al., 2019), there have been several interesting works — (Teney et al., 2020; Krueger et al., 2020; Ahuja et al., 2020; Jin et al., 2020; Chang et al., 2020; Ahuja et al., 2021a; Mahajan et al., 2020; Koyama and Yamaguchi, 2020; Müller et al., 2020; Parascandolo et al., 2021; Ahmed et al., 2021; Robey et al., 2021; Zhang et al., 2021) is an incomplete representative list — that build new methods inspired from IRM to address the OOD generalization problem. We would not go into the details of these different works. However, we believe it is important to talk about works that use conditional independence-based criterion to achieve invariance (Koyama and Yamaguchi, 2020; Huszár, 2019) as those objectives also involve mutual information. Invariance can be enforced using conditional independence as follows. Suppose the environment is given as a random variable E. In this case, if we can learn a representation (X) such that $Y \perp E \mid (X)$, then the predictors learned on are invariant predictors. This conditional independence constraint is formulated in the form of mutual information-based criterion in (Koyama and Yamaguchi, 2020; Huszár, 2019). In this work, we argue that often in classification tasks, there are many representations that satisfy $Y \perp E \mid (X)$ and we have to learn the one that has the least entropy or otherwise OOD generalization is not possible.

A.10.3 Theory of domain adaptation and domain generalization

In the previous section, we discussed works that were directly based on causality/invariance or inspired from it. We now briefly review other relevant works on domain adaptation and domain generalization that are not based on invariance principle from causality. Starting with the seminal works (Ben-David et al., 2007, 2010), there have been many other interesting works in the area of domain adaptation and domain generalization. (Muandet et al., 2013; Zhao et al., 2019; Albuquerque et al., 2019; Piratla et al., 2020; Matsuura and Harada, 2020; Deng et al., 2020; Pagnoni et al., 2018; Greenfeld and Shalit, 2020; Garg et al., 2021) is an incomplete representative list of works that build the theory of domain adaptation and generalization and construct new methods based on it. We recommend the reader to Redko et al. (2019) for further references.

In the case of domain adaptation, many of these works develop bounds on the loss over the target domain using train data and unlabeled target data. In the case of domain generalization, these works develop bounds on the loss over the target domains using training data from multiple domains. Other works (Ben-David and Urner, 2012; David et al., 2010) analyze the minimal conditions under which domain adaptation is possible. In David et al. (2010), the authors showed that the two most common assumptions, a) covariate shifts, and b) the presence of a classifier that achieves close to ideal performance simultaneously in train and test domains, are not sufficient for guaranteed domain adaptation. In this work, we established the necessary and sufficient conditions for domain generalization in linear classification tasks. We showed that under a) covariate shift assumption (SEM in Assumption 2 satisfies the covariate shift), and b) the presence of a common labelling function across all the domains (a much stronger condition than assuming the existence of a classifier that achieves low error across the train and test domains), domain generalization in linear classification is impossible. We showed that adding the requirement that the invariant features satisfy support overlap is both necessary and sufficient (our approach IB-IRM succeeds while IRM and ERM fail) in many cases to guarantee domain generalization.

There has been a long line of research focused on learning domain invariant feature representations (Ganin et al., 2016; Li et al., 2018; Zhao et al., 2020). In these works, the common assumption is that the there exist highly predictive representations whose distributions $P((X^e))$(or distributions conditional on the labels $P((X^e) | Y^e)$) do not change across environments. Note that this is a much stronger assumption than the one typically made in works based on invariance principle (Arjovsky et al., 2019), where the labelling function ($P(Y^e | (X^e))$ does not change. For a detailed analysis of why the assumptions made in these works are too strong and can often fail refer to Arjovsky et al. (2019); Zhao et al. (2019).

A.10.4 Other works on OOD generalization

In Nagarajan et al. (2021) the authors explained why ERM based models trained with gradient descent based approaches fail to generalize OOD in terms of two failure modes – a) gradient descent during training early on relies on shortcut features, b) overparametrized models exhibit geometric biases that cause the models to rely on spurious features. We now describe the line of work based on domain adaptation. For failure mode described in a), we showed in Theorem 5 how information bottleneck penalty can help. Sagawa et al. (2019) studied how overparametrized models can exacerbate the impact of selection biases, Xie et al. (2021) studied the role of auxilliary information and how it can help OOD generalization.

A.10.5 Information bottleneck penalties and impact on generalization

Information bottleneck principle (Tishby et al., 2000) has been used to explain the success of deep learning models; the principle has also been used to build regularizers that can help build models that achieve better in-distribution generalization. We refer the reader to Kirsch et al. (2020), which presents an excellent summary of the existing works on information bottleneck in deep learning. Kirsch et al. (2020) also present a unified framework to view many of the information bottleneck objectives in the literature such as the deterministic information bottleneck (Strouse and Schwab, 2017) and the standard information bottleneck. Other works (Alemi et al., 2016; Arpit et al., 2019) have argued for how information bottleneck can help achieve robustness to adversarial examples, and also to OOD generalization failures. In Arpit et al. (2019), the authors argued that information bottleneck constraints help filter out features that are less correlated with the label. However, the principle of invariance argues for selecting the invariant features even if they have small but invariant correlation with the label over features that maybe strongly correlated but have a varying correlation. As we showed, considering both the principles of invariance and information bottleneck in conjunction is important to achieve OOD generalization (eq. (1)) in a wide range of settings – when the invariant features are fully informative about the label and also when they are partially informative about the label.