Université de Montréal

(Out-of-Distribution?) Generalization in Deep Learning

par

Ethan Caballero

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Mémoire présenté en vue de l'obtention du grade de
Maître ès sciences (M.Sc.)
en informatique

August 31, 2022

Université de Montréal

Faculté des arts et des sciences

Ce mémoire intitulé

(Out-of-Distribution?) Generalization in Deep Learning

présenté par

Ethan Caballero

a été évalué par un jury composé des personnes suivantes :

Esma Aïmeur

(président-rapporteur)

Irina Rish

(directeur de recherche)

Guillaume Rabusseau

(membre du jury)

Résumé

Le principe d'invariance par rapport à la causalité est au cœur d'approches notables telles que la minimisation du risque invariant (IRM) qui cherchent à résoudre les échecs de généralisation hors distribution (OOD). Malgré la théorie prometteuse, les approches basées sur le principe d'invariance échouent dans les tâches de classification courantes, où les caractéristiques invariantes (causales) capturent toutes les informations sur l'étiquette. Ces échecs sont-ils dus à l'incapacité des méthodes à capter l'invariance ? Ou le principe d'invariance lui-même est-il insuffisant ? Pour répondre à ces questions, nous réexaminons les hypothèses fondamentales dans les tâches de régression linéaire, où il a été démontré que les approches basées sur l'invariance généralisent de manière prouvée l'OOD. Contrairement aux tâches de régression linéaire, nous montrons que pour les tâches de classification linéaire, nous avons besoin de restrictions beaucoup plus fortes sur les changements de distribution, sinon la généralisation OOD est impossible. De plus, même avec des restrictions appropriées sur les changements de distribution en place, nous montrons que le principe d'invariance seul est insuffisant. Nous prouvons qu'une forme de contrainte de goulot d'étranglement d'information avec l'invariance aide à résoudre les échecs clés lorsque les caractéristiques invariantes capturent toutes les informations sur l'étiquette et conservent également le succès existant lorsqu'elles ne le font pas. Nous proposons une approche qui combine ces deux principes et démontre son efficacité sur des tests unitaires linéaires [**10**] et sur divers jeux de données réelles de grande dimension.

Mots-clés: Apprentissage en profondeur, généralisation, généralisation hors distribution

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 the 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 combines both these principles and demonstrate its effectiveness on linear unit tests [**10**] and on various high-dimensional real datasets.

Keywords: Deep Learning, Generalization, Out-of-Distribution Generalization

Contents

Résumé 5

Abstract 7

List of tables 11

List of figures 13

List of acronyms and abbreviations 15

Acknowledgements 17

Chapter 1. Introduction 19

1.1. Structural Equation Model Example 20

1.1.1. Structural equation models and assumptions on $\mathcal{E}_{all}$ 20

1.2. IRM background 21

1.3. Related works 23

1.3.1. Invariance principles in causality 23

1.3.2. Invariance principles in OOD generalization 24

1.3.3. Theory of domain adaptation and domain generalization 25

1.3.4. Other works on OOD generalization 25

1.3.5. Information bottleneck penalties and impact on generalization 26

Chapter 2. Invariance Principle Meets Information Bottleneck for Out-of-Distribution Generalization 27

2.1. Introduction 28

2.2. OOD generalization and invariance: background & failures 29

2.3. OOD generalization theory for linear classification tasks 32

2.4. Information bottleneck principle meets invariance principle 36

2.4.1. Proposed approach 38

2.5. Experiments 39

2.6. Extensions, limitations, and future work 40

2.7. Conclusion 42

Acknowledgements 42

Funding disclosure 42

2.8. Appendix 43
2.8.1. Societal impact 43
2.8.2. Experiments details 43
2.8.2.1. Datasets 43
2.8.2.2. Training and evaluation procedure 46
2.8.2.3. Supplementary experiments 48
2.8.2.4. Compute description 50
2.8.2.5. Assets used and the license details 51
2.8.3. Remark on the linear general position assumption and its implications on support overlap 52
2.8.4. Notations and proof of Theorem 2 (impossibility of guaranteed OOD generalization for linear classification) 54
2.8.5. Proof of Theorem 3: sufficiency and insufficiency of ERM and IRM 60
2.8.6. Proof of Theorem 4: IB-IRM and IB-ERM vs. IRM and ERM 64
2.8.6.1. Discrete random variables 66
2.8.6.2. Continuous random variables 70
2.8.7. Derivation of the final objective in equation (2.4.2) 76
2.8.8. Proof of Theorem 5: impact of IB on the learning speed 79
2.8.9. Illustrating both invariance and information bottleneck acting in conjunction 83

Chapter 3. Conclusion 87

References 89

List of tables

2.1 Summary of the new and existing results [**7, 55**]. IB-ERM (IRM): information bottleneck - empirical (invariant) risk minimization ERM (IRM)................. 29

2.2 Categorization of OOD evaluation datasets and SEMs. Example 1/1S, 2/2S, 3/3S from [**10**], AC-CMNIST[**7**], CS-CMNIST[**4**]. 32

2.3 Comparisons on linear unit tests in terms of mean square error (regression) and classification error (classification). "#Envs" means the number of training environments.. 41

2.4 Classification accuracy percentage on colored MNISTs, Terra Incognita and COCO dataset.. 41

2.5 Hyperparameters distributions for random search given included penalty of the algorithm.. 48

2.6 AC-CMNIST. Comparisons of the methods using the train-domain validation set tuning procedure [**25**]. The percentages in the columns indicate what fraction of the total data (50000 points) is used for validation.............................. 49

2.7 CS-CMNIST. Comparisons of the methods using the oracle test-domain validation set tuning procedure [**25**]. The percentages in the columns indicate what fraction of the total data (50000 points) is used for validation.......................... 49

2.8 CS-CMNIST. Comparisons of the methods using the train-domain validation set tuning procedure [**25**]. The percentages in the columns indicate what fraction of the total data (50000 points) is used for validation.............................. 49

2.9 CS-CMNIST. Comparisons of the methods using the oracle test-domain validation set tuning procedure [**25**]. The percentages in the columns indicate what fraction of the total data (50000 points) is used for validation.......................... 50

List of figures

2.1 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 $\mathcal{E}_{tr}$, 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. 33

2.2 Comparison of the DAG from Assumption 2 (fully informative invariant features) vs. DAGs from [**55, 7**] (partially informative invariant features). 37

2.3 Comparing convergence of $\frac{|w_{\mathsf{spu}}|}{\sqrt{w_{\mathsf{spu}}^2+w_{\mathsf{inv}}^2}}$ (metric from [**42**]) for average selection bias $p = 0.9$. 38

2.4 Illustrating the impact of the IB and IRM penalty on linear unit tests [**10**] 50

List of acronyms and abbreviations

ML	Machine Learning
ERM	Empirical Risk Minimization
IRM	Invariant Risk Minimization
OOD	Out-of-Distribution
i.i.d.	Independent and Identically Distributed
IB	Information Bottleneck
SEM	Structural Equation Model
DNN	Deep Neural Network
AGI	Artificial General Intelligence
MNIST	Modified National Institute of Standards and Technology (dataset)

COCO	Common Objects in Context (dataset)

Acknowledgements

I would like to thank all sentient beings.

Chapter 1

Introduction

Machine Learning is a set of methods for getting machines to learn from data. The purpose of this learning is usually to perform tasks that people find to be economically valuable. Ideally, we want the machine(s) to learn things (e.g. representations) that generalize to data that is nonidentical (in raw data (e.g. pixel) space) to previous data the machine has learned from because most of the real configurations of raw data space happen only once.

There are a few paradigms (such as supervised learning, unsupervised learning, and reinforcement learning) by which this learning can take place, but for now we will focus on supervised learning to elucidate how the topic of generalization manifests.

Supervised learning is the task of learning a function hypothesis $f \in H$ from a hypothesis class H that map inputs to outputs: $X \rightarrow Y$. In this problem setup, the machine learning model is usually provided inputs $x \in X$ and outputs $y \in Y$, where each pair (x, y) is drawn from an unknown joint distribution, D. Given a loss function ℓ, typical supervised learning evaluates the performance of a predictor via the expected loss (also known as the risk): $R(f) = \mathbb{E}_{(x,y)\sim D}\Big[\ell(f(x), y)\Big]$.

In practice, only a finite number of samples S (usually) from D can be trained on, so what ends up being directly minimized while learning is the empirical risk: $\hat{R}(f) = \mathbb{E}_{(x,y)\sim S}\Big[\ell(f(x), y)\Big]$. This learning via minimization of empirical risk is formally called empirical risk minimization (ERM).

One way we usually measure the generalization ability of f is via the generalization gap: $R(f) - \hat{R}(f)$. Often, it is assumed that the samples S used to minimize $\hat{R}(f)$ are sampled i.i.d. (independent and identically distributed) from distribution D that we are evaluating performance on via $R(f)$. When this assumption is met, we refer to the generalization gap as the in-distribution (or out-of-sample) generalization gap. However, in practice this assumption often is not met due to reasons such as selection bias and confounding.

A more ambitious form of generalization referred to as out-of-distribution (OOD) generalization seeks to find function f that best generalizes to the set of distributions (each of which is also sometimes referred to as a dataset D) from all possible environments $e \in \mathcal{E}_{all}$. We use the word "possible" in the modal realist sense that invokes all possible worlds [**35**]; for example, one could intervene such that the sun is removed [**7**]. In this setting, data D is generated from a set of environments $\mathcal{E}_{all}$: $D = \{D^e\}_{e \in \mathcal{E}_{all}}$, where $D^e = \{x_i^e, y_i^e\}_{i=1}^{n^e}$ is the dataset from environment $e \in \mathcal{E}_{all}$ and n^e is the number of instances in environment e. The risk in each individual environment is $R^e(f) = \mathbb{E}_{(x^e, y^e) \sim D^e}\Big[\ell(f(x^e), y^e)\Big]$. Formally stated, the goal of the OOD generalization problem statement is to find the predictor which minimizes the following minimax:

$$\min_f \max_{e \in \mathcal{E}_{all}} R^e(f). \tag{1.0.1}$$

The solution to this minimax is the function that satisfies these two criteria:

(1) the function only uses features of X that are causes of Y to predict Y

(2) the function is the most predictive (of Y) mapping from X to Y that also meets criterion 1

We now present an example SEM (Structural Equation Model) [**46**] which shows how satisfying these two criteria yields the solution to the minimax in equation (1.0.1), which is borrowed from [**7**].

1.1. Structural Equation Model Example

1.1.1. Structural equation models and assumptions on $\mathcal{E}_{all}$

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

$$\mathcal{S}_i : X_i \leftarrow f_i(\mathsf{Pa}(X_i), N_i), \tag{1.1.1}$$

where $\mathsf{Pa}(X_i)$ 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 $X_j \in \mathsf{Pa}(X_i)$. 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 $\mathcal{C}$ is the process of replacing one or several of its structural equations to obtain a new intervened SEM $\mathcal{C}^e = (\mathcal{S}^e, N^e)$, with structural equations given as*

$$\mathcal{S}_i^e : X_i^e \leftarrow f_i^e(\mathsf{Pa}(X_i^e), N_i^e), \tag{1.1.2}$$

where the variable X_i^e is said to be intervened if $\mathcal{S}_i \neq \mathcal{S}_i^e$ or $N_i \neq N_i^e$

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

Definition 3. *Consider a SEM $\mathcal{C}$ 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 $\mathcal{E}_{all}(\mathcal{C})$ indexes all the interventional distributions $\mathbb{P}^e$, where $(X^e,Y^e) \sim \mathbb{P}^e$. An intervention e is valid if the following conditions are met: i) the causal graph remains acyclic, ii) $\mathbb{E}[Y^e|\mathsf{Pa}(Y)] = \mathbb{E}[Y|\mathsf{Pa}(Y)]$, i.e. expectation conditional on parents is invariant, and the variance $\mathsf{Var}[Y^e|\mathsf{Pa}(Y)]$ remains within a finite range.*

Following the above definitions it is possible to show that a predictor that relies on causal parents only $v : \mathbb{R}^d \rightarrow \mathcal{Y}$ and is given as $v(x) = \mathbb{E}[f_Y(\mathsf{Pa}(Y), N_Y)]$ solves the OOD generalization problem in equation (1.0.1) over the environments $\mathcal{E}_{all}(\mathcal{C})$ 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 [**7**] to explain why models that rely on causes are more robust to valid interventions $\mathcal{E}_{all}$ discussed in the previous section.

$$\begin{aligned} Y^e &\leftarrow X^e_{\mathsf{inv}} + \epsilon^e \\ X^e_{\mathsf{spu}} &\leftarrow Y^e + \zeta^e \end{aligned} \tag{1.1.3}$$

where $X^e_{\mathsf{inv}} \in \mathcal{N}(0,(\sigma^e)^2)$ is the cause of Y^e, $\epsilon^e \in \mathcal{N}(0,(\sigma^e)^2)$ is noise, X^e_{spu} is the effect of Y^e and $\zeta^e \in \mathcal{N}(0, \sigma)$ is also noise. Suppose there are two training environments $\mathcal{E}_{tr} = \{e_1,e_2\}$, in the first $(\sigma^{e_1})^2 = 1$ and in the second $(\sigma^{e_2})^2 = 2$. In each of every environment, $\sigma^2 = 1$. The three possible models $w_{\mathsf{inv}}X^e_{\mathsf{inv}} + w_{\mathsf{spu}}X^e_{\mathsf{spu}}$ we could build are as follows: a) regress only on X^e_{inv}, then in the optimal model $w_{\mathsf{inv}} = 1, w_{\mathsf{spu}} = 0$, b) regress only on X^e_{spu} and get $w_{\mathsf{inv}} = 0, w_{\mathsf{spu}} = \frac{\sigma^2}{(\sigma^e)^2+\frac{1}{2}}$, c) regress on $(X^e_{\mathsf{inv}}, X^e_{\mathsf{spu}})$ to get $w_{\mathsf{inv}} = \frac{1}{(\sigma^e)^2+1}$ and $w_{\mathsf{spu}} = \frac{(\sigma^e)^2}{(\sigma^e)^2+1}$. Observe that the predictor that focuses on the cause only does not depend on σ^2 and is thus invariant to distribution shifts induced by change in $(\sigma^e)^2$, which is not the case with the other models. For environments in $\mathcal{E}_{all}$ we can change the distribution of X^e_{inv} and X^e_{spu} arbitrarily. Consider an environment $e \in \mathcal{E}_{all}$ where X^e_{spu} 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^e_{inv} does not change.

1.2. IRM background

IRM (Invariant Risk Minimization) [**7**] proposed one way to address the OOD Generalization problem statement. In IRM, the training data is gathered from multiple environments. The set of training environments is defined as $\mathcal{E}_{tr}$. Define the training dataset $D = \{D_e\}_{e\in\mathcal{E}_{tr}}$, where $D_e = \{\boldsymbol{x}^i_e, y^i_e\}_{i=1}^{n_e}$ is the dataset gathered from environment $e \in \mathcal{E}_{tr}$ and n_e is the number of points in environment e. $\boldsymbol{x}^i_e \in \mathcal{X}$ and $y^i_e \in \mathcal{Y}$ correspond to the feature value for i^{th} data point and the label for i^{th} data point respectively. Each $(\boldsymbol{x}^i_e,y^i_e)$ is an i.i.d. draw from $\mathbb{P}_e$. IRM's objective is to use these datasets D to construct a predictor $f : \mathcal{X} \rightarrow \mathbb{R}$ that performs well across many environments $\mathcal{E}_{all}$, where $\mathcal{E}_{tr} \subset \mathcal{E}_{all}$. Define the risk of f in environment e

as $R_e(f) = \mathbb{E}_e\Big[\ell(f(\boldsymbol{X}_e), Y_e)\Big]$, where ℓ can be cross-entropy loss, error rate, square loss, and $(\boldsymbol{X}_e, Y_e) \sim \mathbb{P}_e$ and the expectation $\mathbb{E}_e$ is defined w.r.t to distribution of points in environment e. Formally stated the goal is to solve

$$\min_f \max_{e \in \mathcal{E}_{all}} R_e(f) \tag{1.2.1}$$

In the above problem, we have not stated any restrictions on $\mathcal{E}_{all}$. Without any restriction on $\mathcal{E}_{all}$ it is easy to see that we can always construct an unseen environment adversarially that ensures that any method has error rate of one. Suppose a method uses the training environments and learns a function f^*_{tr}; the adversary can use this function f^*_{tr} to create a test environment with labels based on $1 - f^*_{tr}$. This shows that the set of environments $\mathcal{E}_{all}$ need to be restricted in a meaningful manner.

We describe the approach taken by [**7**] to restrict the environments. Assume that the data across all the environments is governed by a family of structural equation models (SEMs) defined as follows, for each variable $W \in \{\boldsymbol{X}^1, \dots, \boldsymbol{X}^d\} \cup Y_e$:

$$W \leftarrow f_W(\mathsf{Pa}(W), \epsilon), \tag{1.2.2}$$

where f_W is a map from the feature space to the domain of the corresponding random variable and ϵ is noise. An intervention is defined as the modification to the SEM, i.e. for at least one of the variables W the SEM is modified, i.e. either f_W is changed or ϵ is changed. Each environment is represented by an intervention and $\mathcal{E}_{all}$ contains all the interventions except the ones in which the variable Y has been intervened on. Having defined constraints on the environments, we now describe a solution to the problem in equation 1.0.1. We make a few more assumptions: (a) $Y \leftarrow f_Y(\mathsf{Pa}(Y)) + \epsilon$ the SEM of Y has additive noise and the noise is zero mean and variance in the noise is bounded; (b) $\exists$ a map $\Phi^* : \mathcal{X} \rightarrow \mathcal{H}$, which we call an *invariant feature map*, such that $\mathbb{E}\Big[Y^e\Big|\Phi^*\Big(X^e\Big)\Big]$ is the same for all $e \in \mathcal{E}_{all}$ and $Y^e \not\perp \Phi^*(X^e)$; (c) the set of parents $\mathsf{Pa}(Y) = \Phi^*(\boldsymbol{X})$; (d) $\exists$ an environment $e \in \mathcal{E}_{all}$ where $Y^e \perp X^e|\Phi^*(X^e)$ [**4**]. Under these assumptions $f_Y(\Phi^*(\boldsymbol{X}))$ solves the problem in equation (1.0.1). The objective of IRM that we describe in the next section tries to solve for $f_Y(\Phi^*(\boldsymbol{X}))$.

In the entire description above, we took the assistance of SEMs. However, we can state the above requirements in a more general way and require that $\mathbb{E}_e[Y_e|\Phi^*(\boldsymbol{X}_e)]$ is invariant across environments.

Invariant predictor and IRM optimization: An invariant predictor is composed of two parts: a representation map and a classifier. Define a representation map $\Phi : \mathcal{X} \rightarrow \mathcal{H}$ (that transforms X^e as $\Phi(X^e)$) and define a linear classifier $w : \mathcal{H} \rightarrow \mathcal{Y}$ (that operates on the representation as $w \circ \Phi(X^e)$).

We want to search for representation map Φ such that $\mathbb{E}[Y^e|\Phi(X^e)]$ is invariant. We say that a representation map Φ elicits an invariant predictor $w \circ \Phi$ across the set of training environments $\mathcal{E}_{tr}$ if there is a predictor w that simultaneously achieves the minimum risk, i.e., $w \in \arg\min_{\tilde{w}} R^e(\tilde{w} \circ \Phi),\ \forall e \in \mathcal{E}_{tr}$. The main objective of IRM is stated as

$$\min_{w:\mathcal{H}\to\mathcal{Y},\Phi:\mathcal{X}\to\mathcal{H}} \frac{1}{|\mathcal{E}_{tr}|} \sum_{e\in\mathcal{E}_{tr}} R^e(w \circ \Phi) \quad \text{s.t. } w \in \arg\min_{w:\mathcal{H}\to\mathcal{Y}} R^e(\tilde{w} \circ \Phi), \forall e \in \mathcal{E}_{tr}. \tag{1.2.3}$$

Note that [**7**] and others sometimes use a simplified notation in which Φ (and $\Phi(X)$) represents the hidden state output by the representation map and w represents the parameter vector of the linear classifier on the top of the hidden state; throughout the rest of this thesis, this simplified notation is sometimes used.

Define the set of invariant predictors $\boldsymbol{w} \cdot \boldsymbol{\Phi}$ satisfying the constraints in (1.2.3) as $\mathcal{S}^{\mathsf{IV}}$. Informally stated, the main idea behind the above optimization is inspired from invariance principles in causality [**11**][**46**]. Each environment can be understood as an intervention. By learning an invariant predictor the learner hopes to identify a representation $\boldsymbol{\Phi}$ that transforms the observed features into the causal features, and the optimal model trained on causal representations is likely to be same (invariant) across the environments provided we do not intervene on the label itself. These invariant models can be shown to have a good out-of-distribution performance.

The authors of [**7**] also propose a more practical algorithm called IRMv1 for solving the IRM problem:

$$\min_{\boldsymbol{\Phi}\in\mathbb{R}^{r\times d}} \sum_{e\in\mathcal{E}_{tr}} R_e(\boldsymbol{\Phi}(X^e)) + \lambda \|\nabla_{\boldsymbol{w}} R_e(\boldsymbol{w} \cdot \boldsymbol{\Phi}(X^e)\|_2^2 \tag{1.2.4}$$

in which $\boldsymbol{w}$ is a fixed vector of ones, and λ controls the weighting of the penalty term on the gradients on $\boldsymbol{w}$.

1.3. Related works

1.3.1. Invariance principles in causality

The foundations of invariance principles are rooted in the theory of causality [**45**]. 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 $\mathbb{P}(Y|X_{\mathrm{Pa}(Y)})$ remains invariant. Related and similar notions are *stability* [**46**], *autonomy* [**57**], *invariant causal prediction principle* [**48, 26**]. 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 ($\Phi(X) = X_{\mathrm{Pa}(y)}$) to predict Y is invariant under interventions.

1.3.2. 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 [**48, 7**], 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 [**53**] 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 [**33**]. 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? [**7**] 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. [**55**] 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.

Recent works [**55, 54, 29, 25**] have pointed to several limitations of invariance based approaches for addressing OOD generalization failures. In [**55**], 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 [**37**], 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 [**31**] 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 [**37**] 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 [**29**], 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.

Methods. Following the original works ICP (Invariant Causal Prediction) [**48**] and IRM [**7**], there have been several interesting works — [**61, 34, 3, 28, 16, 2, 38, 33, 41, 44,**

1, 52, 66] 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 [**33, 27**]. 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 $\Phi(X)$ such that $Y \perp E|\Phi(X)$, then the predictors learned on Φ are invariant predictors. This conditional independence constraint is formulated in the form of mutual information-based criterion in [**33, 27**].

1.3.3. 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 [**14, 13**], there have been many other interesting works in the area of domain adaptation and domain generalization. [**40, 67, 5, 50, 39, 20, 43, 24, 22**] 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 [**51**] 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 [**15, 18**] analyze the minimal conditions under which domain adaptation is possible. In [**18**], 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.

There has been a long line of research focused on learning domain invariant feature representations [**21, 36, 68**]. In these works, the common assumption is that the there exist highly predictive representations whose distributions $\mathbb{P}(\Phi(X^e))$(or distributions conditional on the labels $\mathbb{P}(\Phi(X^e)|Y^e)$) do not change across environments. Note that this is a much stronger assumption than the one typically made in works based on invariance principle [**7**], where the labelling function ($\mathbb{P}(Y^e|\Phi(X^e))$ does not change. For a detailed analysis of why the assumptions made in these works are too strong and can often fail refer to [**7, 67**].

1.3.4. Other works on OOD generalization

In [**42**] 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. [**56**] studied how overparametrized models can exacerbate the impact of selection biases, [**65**] studied the role of auxilliary information and how it can help OOD generalization.

1.3.5. Information bottleneck penalties and impact on generalization

Information bottleneck principle [**62**] 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. In short, the information bottleneck principle says that we should prefer the predictor f that has lower $I(X;\Phi(X))$ (mutual information between X and $\Phi(X)$). We refer the reader to [**32**], which presents an excellent summary of the existing works on information bottleneck in deep learning. [**32**] also present a unified framework to view many of the information bottleneck objectives in the literature such as the deterministic information bottleneck [**60**] and the standard information bottleneck. Other works [**6, 8**] have argued for how information bottleneck can help achieve robustness to adversarial examples, and also to OOD generalization failures. In [**8**], 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 show in the next section, considering both the principles of invariance and information bottleneck in conjunction is important to achieve OOD generalization (eq. (1.0.1)) in a wide range of settings.

Chapter 2

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

Authors: Kartik Ahuja† and Ethan Caballero*† and Dinghuai Zhang[1] *† and Yoshua Bengio† and Ioannis Mitliagkas† and Irina Rish[2]

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 combines both these principles and demonstrate its effectiveness on linear unit tests [**10**] and on various high-dimensional real datasets.

Contribution: I designed, implemented, and ran most of the experiments. I wrote most of the Experiments section of paper. I co-designed the practical version of the information bottleneck penalty. I implemented, ran, and co-conceived high-dimensional real-world

[1] * means equal contribution

[2] † means affiliation is Mila - Quebec AI Institute, Université de Montréal, Quebec, Canada.

experiments for the NeurIPS rebuttal. There are two github repositories: A final github repositories (`https://github.com/ahujak/IB-IRM`) posted by Kartik Ahuja for the official release and a github repository (`https://github.com/ethancaballero/ib_irm`) posted by me that contains most of my code contributions during development of this work.

Submission: This work is accepted as conference track paper (and conference spotlight presentation) of NeurIPS 2021.

2.1. Introduction

Recent years have witnessed an explosion of examples showing deep learning models are prone to exploiting shortcuts (spurious features) [**23, 49**] which make them fail to generalize out-of-distribution (OOD). In [**12**], 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 [**19**]).

To address these out-of-distribution generalization failures, invariant risk minimization [**7**] and several other works were proposed [**3, 49, 34, 52, 66**]. The invariance principle from causality [**47, 45**] 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 [**53, 33, 4**] under many distribution shifts but the same is not true for other predictors.

Our contributions. Despite the promising theory, invariance principle-based approaches fail in settings [**10**] where invariant features capture all information about the label contained in the input. A particular example is image classification (e.g., cow vs. camel) [**12**] 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 [**62**], together with the invariance principle, provably works in both settings – when invariant features completely capture the information about

[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.

the label and also when they do not. (Table 2.1 summarizes our theoretical results presented later). We propose an approach that combines both these principles and demonstrate its effectiveness on linear unit tests [**10**] and on various high-dimensional real datasets.

Task	Invariant features capture label info	Support overlap invariant features	Support overlap spurious features	**OOD generalization guarantee** ($\mathcal{E}_{tr} \to \mathcal{E}_{all}$)			
				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 2.1. Summary of the new and existing results [**7, 55**]. IB-ERM (IRM): information bottleneck - empirical (invariant) risk minimization ERM (IRM).

2.2. OOD generalization and invariance: background & failures

Background. We consider a supervised training data D gathered from a set of training environments $\mathcal{E}_{tr}$: $D = \{D^e\}_{e\in\mathcal{E}_{tr}}$, where $D^e = \{x_i^e, y_i^e\}_{i=1}^{n^e}$ is the dataset from environment $e \in \mathcal{E}_{tr}$ and n^e is the number of instances in environment e. $x_i^e \in \mathbb{R}^d$ and $y_i^e \in \mathcal{Y} \subseteq \mathbb{R}^k$ 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 $\mathbb{P}^e$, where $\mathbb{P}^e$ is the joint distribution of the input feature and the label in environment e. Let $\mathcal{X}^e$ 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 $\mathcal{E}_{all}$, where $\mathcal{E}_{all} \supset \mathcal{E}_{tr}$. Define the risk of f in environment e as $R^e(f) = \mathbb{E}\Big[\ell(f(X^e), Y^e)\Big]$, where for example ℓ can be 0-1 loss, logistic loss, square loss, $(X^e, Y^e) \sim \mathbb{P}^e$, and the expectation $\mathbb{E}$ is w.r.t. $\mathbb{P}^e$. Formally stated, our goal is to use the data from training environments $\mathcal{E}_{tr}$ to find $f : \mathbb{R}^d \to \mathcal{Y}$ to minimize

$$\min_f \max_{e\in\mathcal{E}_{all}} R^e(f). \tag{2.2.1}$$

So far we did not state any restrictions on $\mathcal{E}_{all}$. Consider binary classification: without any restrictions on $\mathcal{E}_{all}$, no method can reduce the above objective (ℓ is 0-1 loss) to below one. Suppose a method outputs f^*; if $\exists\ e \in \mathcal{E}_{all} \setminus \mathcal{E}_{tr}$ with labels based on $1 - f^*$, then it achieves an error of one. Some assumptions on $\mathcal{E}_{all}$ are thus necessary. Consider how $\mathcal{E}_{all}$ is restricted using invariance for linear regressions [**7**].

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

$$\begin{aligned} &Y^e \leftarrow w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \epsilon^e, \quad Z^e_{\mathsf{inv}} \perp \epsilon^e, \quad \mathbb{E}[\epsilon^e] = 0, \mathbb{E}\Big[|\epsilon^e|^2\Big] \leq \sigma^2_{\mathsf{sup}} \\ &X^e \leftarrow S(Z^e_{\mathsf{inv}}, Z^e_{\mathsf{spu}}) \end{aligned} \tag{2.2.2}$$

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

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 ϵ^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 ϵ^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 ϵ^e (σ^2_{sup}). If we regress on the entire Z^e and the optimal predictor places a non-zero weight on Z^e_{spu} (e.g., $Z^e_{\mathsf{spu}} \leftarrow Y^e + \zeta^e$), then this predictor fails to solve equation (2.2.1) ($\exists\ e \in \mathcal{E}_{all}$, $Z^e_{\mathsf{spu}} \to \infty$, error $\to \infty$, 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 (2.2.1) under Assumption 1 and achieves a value of σ^2_{sup} for the objective in equation (2.2.1).

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

$$\min_{w\in\mathbb{R}^{k\times r},\Phi\in\mathbb{R}^{r\times d}} \frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}} R^e(w\cdot\Phi) \quad \text{s.t. } w \in \arg\min_{\tilde{w}\in\mathbb{R}^{k\times r}} R^e(\tilde{w}\cdot\Phi), \forall e \in \mathcal{E}_{tr}. \tag{2.2.3}$$

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) [**64**] (assuming all the training environments occur with equal probability). In all our theorems, we use 0-1 loss for binary classification $\mathcal{Y} = \{0{,}1\}$ and square loss for regression $\mathcal{Y} = \mathbb{R}$. For binary classification, the output of the predictor is given as $\mathsf{I}(w\cdot\Phi(X^e))$, where $\mathsf{I}(\cdot)$ is the indicator function that takes 1 if the input is ≥ 0 and 0 otherwise, and the risk is $R^e(w\cdot\Phi) = \mathbb{E}\Big[\big|\mathsf{I}(w\cdot\Phi(X^e)) - Y^e\big|\Big]$. For regression, the output of the predictor is $w\cdot\Phi(X^e)$ and the corresponding risk is $R^e(w\cdot\Phi) = \mathbb{E}\Big[(w\cdot\Phi(X^e) - Y^e)^2\Big]$. We now present the main OOD generalization result from [**7**] for linear regressions.

Theorem 1. *(Informal) If Assumption 1 is satisfied,* $\mathsf{Rank}[\Phi] > 0$*, $|\mathcal{E}_{tr}| > 2d$, and $\mathcal{E}_{tr}$ lie in a linear general position (a mild condition on the data in $\mathcal{E}_{tr}$, defined in the Appendix), then each solution to equation* (2.2.3) *achieves OOD generalization (solves equation* (2.2.1)*, $\nexists\ e \in \mathcal{E}_{all}$ with risk $> \sigma^2_{\mathsf{spu}}$).*

[4] In many examples in the literature, invariant features are causal, but not always [**55**].

Despite the above guarantees, IRM has been shown to fail in several cases including linear SEMs in [**10**]. 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 $e \in \mathcal{E}_{all}$, the distribution $(X^e,Y^e) \sim \mathbb{P}^e$ satisfies the following properties. a) $\exists$ a map Φ^* (linear or not), which we call an *invariant feature map*, such that $\mathbb{E}\Big[Y^e\Big|\Phi^*\Big(X^e\Big)\Big]$ is the same for all $e \in \mathcal{E}_{all}$ and $Y^e \not\perp \Phi^*(X^e)$. These conditions ensure Φ^* maps to features that have a finite predictive power and have the same optimal predictor across $\mathcal{E}_{all}$. For the SEM in Assumption 1, Φ^* maps to Z^e_{inv}. b) $\exists$ a map Ψ^* (linear or not), which we call *spurious feature map*, such that $\mathbb{E}\Big[Y^e\Big|\Psi^*\Big(X^e\Big)\Big]$ is not the same for all $e \in \mathcal{E}_{all}$ and $Y^e \not\perp \Psi^*(X^e)$ for some environments. Ψ^* often creates a hindrance in learning predictors that only rely on Φ^*. Note that Ψ^* should not be a transformation of some Φ^*. For the SEM in Assumption 1, suppose Z^e_{spu} is anti-causally related to Y^e, then Ψ^* maps to Z^e_{spu} (See Appendix for an example).

In the colored MNIST (CMNIST) dataset [**7**], 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 Φ^* extracts the uncolored digit and the spurious feature map Ψ^* extracts the background color. [**4**] 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 [**7**]) but is no different from ERM and fails in covariate shift-CMNIST (CS-CMNIST). In AC-CMNIST, the invariant features $\Phi^*(X^e)$ (uncolored digit) are *partially informative* about the label, i.e., $Y \not\perp X^e|\Phi^*(X^e)$, and color contains information about label not contained in the uncolored digit. On the other hand in CS-CMNIST, invariant features are *fully informative* about the label, i.e., $Y \perp X^e|\Phi^*(X^e)$, 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 2.2, 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 [**4**]; this is consistent with the linear regression result in Theorem 1, where IRM succeeds regardless of whether $Y^e \perp X^e|Z^e_{\mathsf{inv}}$ holds

[5] The deterministic labelling case was referred as realizable problems in [**7**].

Fully informative invariant features (FIIF)	**Partially informative invariant features (PIIF)**
$\forall e \in \mathcal{E}_{all}, Y^e \perp X^e \vert \Phi^*(X^e)$	$\exists\ e \in \mathcal{E}_{all}\ Y^e \not\perp X^e \vert \Phi^*(X^e)$
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 [**55**]
ERM and IRM fail	**ERM fails, IRM succeeds sometimes**
Theorem 3,4 (This paper)	Theorem 9, 5.1 [**7, 55**]

Table 2.2. Categorization of OOD evaluation datasets and SEMs. Example 1/1S, 2/2S, 3/3S from [**10**], AC-CMNIST[**7**], CS-CMNIST[**4**].

or not. Motivated by this observation, we take a closer look at the classification tasks where invariant features are fully informative.

2.3. 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 [**42**]), which can be understood as a simplified version of the CS-CMNIST dataset [**4**], Example 2/2S of [**10**], where both IRM and ERM fail. The example goes as follows. In each training environment $e \in \mathcal{E}_{tr}$

$$\begin{aligned}
&Y^e \leftarrow \mathsf{I}\Big(X^e_{\mathsf{inv}} - \frac{1}{2}\Big), \text{ where } X^e_{\mathsf{inv}} \in \{0,1\} \text{ is } \mathsf{Bernoulli}\Big(\frac{1}{2}\Big),\\
&X^e_{\mathsf{spu}} \leftarrow X^e_{\mathsf{inv}} \oplus W^e, \text{ where } W^e \in \{0,1\} \text{ is } \mathsf{Bernoulli}\Big(1-p^e\Big) \text{ with selection bias } p^e > \frac{1}{2},
\end{aligned} \tag{2.3.1}$$

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 $\mathcal{E}_{tr}$ to $\mathcal{E}_{all}$, 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 2.1a) for a pictorial representation of this example illustrating the gist of the problem: there are many classifiers with the same error on $\mathcal{E}_{tr}$ while only the one identical to the labelling function $\mathsf{I}(X^e_{\mathsf{inv}} - \frac{1}{2})$ generalizes correctly OOD. Define a classifier $\mathsf{I}(w_{\mathsf{inv}}x_{\mathsf{inv}} + w_{\mathsf{spu}}x_{\mathsf{spu}} - \frac{1}{2}(w_{\mathsf{inv}} + w_{\mathsf{spu}}))$. Define a set of classifiers $\mathcal{S} = \{(w_{\mathsf{inv}}, w_{\mathsf{spu}}) \text{ s.t. } w_{\mathsf{inv}} > |w_{\mathsf{spu}}|\}$. Observe that all the classifiers in $\mathcal{S}$ achieve a zero classification error on the training environments. However, only classifiers for which $w_{\mathsf{spu}} = 0$ solve the OOD generalization (eq. (2.2.1)). With Φ as the identity, it can be shown that all the classifiers $\mathcal{S}$ form an invariant predictor (satisfy the constraint in equation (2.2.3) 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 [**7**],

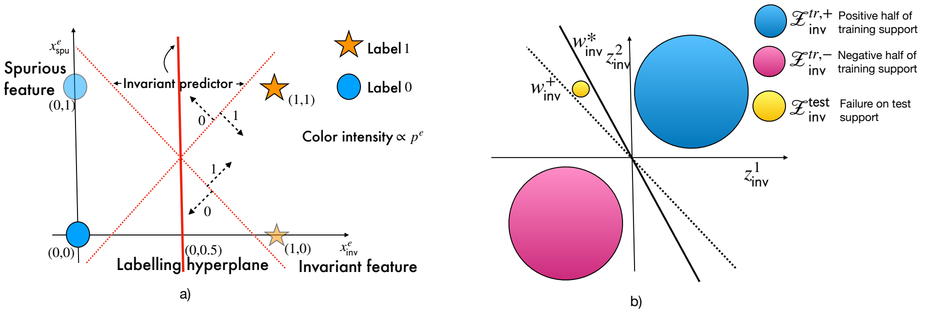


Fig. 2.1. a) 2D classification example illustrating multiple invariant predictors: Most of these predictors rely on spurious features and each of them achieve zero error across all $\mathcal{E}_{tr}$, b) illustration of the impossibility result. If latent invariant features in the training environments are separable, then there are multiple equally good candidates that could have generated the data, and the algorithm cannot distinguish between these.

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. (1.0.1)). [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.

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

$$\begin{aligned} &Y^e \leftarrow \mathsf{I}\Big(w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}}\Big) \oplus N^e, \qquad N^e \sim \mathsf{Bernoulli}(q), q < \frac{1}{2}, \qquad N^e \perp (Z^e_{\mathsf{inv}}, Z^e_{\mathsf{spu}}), \\ &X^e \leftarrow S\Big(Z^e_{\mathsf{inv}}, Z^e_{\mathsf{spu}}\Big), \end{aligned} \tag{2.3.2}$$

where $w^*_{\mathsf{inv}} \in \mathbb{R}^m$ *with* $\|w^*_{\mathsf{inv}}\| = 1$ *is the labelling hyperplane,* $Z^e_{\mathsf{inv}} \in \mathbb{R}^m$*,* $Z^e_{\mathsf{spu}} \in \mathbb{R}^o$*,* 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 2.2a) for the directed acyclic graph (DAG) corresponding to this SEM. From the DAG observe that $Y^e \perp X^e | Z^e_{\mathsf{inv}}$, which implies that the invariant features are fully informative. Contrast this with a DAG that follows Assumption 1 shown in Figure 2.2b), where $Y^e \not\perp X^e | Z^e_{\mathsf{inv}}$ and thus the invariant features are not fully informative. If $\mathcal{E}_{all}$ 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 $\mathsf{I}(w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}})$ can solve the OOD generalization (eq. (2.2.1)); such a classifier achieves an error of q (noise level) in all the environments. As a result, if for a classifier we can find $e \in \mathcal{E}_{all}$ that follows Assumption 2 where the error is greater than q, then such a classifier does not solve equation (2.2.1).

[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).

Now we ask – what are the minimal conditions on training environments $\mathcal{E}_{tr}$ to achieve OOD generalization when $\mathcal{E}_{all}$ 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. (2.2.1))? 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 $\mathcal{Z}^e_{\mathsf{inv}}$ ($\mathcal{Z}^e_{\mathsf{spu}}$).

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

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

Assumption 5. ***Invariant feature support overlap.*** $\forall e \in \mathcal{E}_{all}, \mathcal{Z}^e_{\mathsf{inv}} \subseteq \cup_{e'\in\mathcal{E}_{tr}}\mathcal{Z}^{e'}_{\mathsf{inv}}$

Assumption 6. ***Spurious feature support overlap.*** $\forall e \in \mathcal{E}_{all}, \mathcal{Z}^e_{\mathsf{spu}} \subseteq \cup_{e'\in\mathcal{E}_{tr}}\mathcal{Z}^{e'}_{\mathsf{spu}}$

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 $\mathsf{Inv\text{-}Margin} = \min_{z\in\cup_{e\in\mathcal{E}_{tr}}\mathcal{Z}^e_{\mathsf{inv}}} \mathsf{sgn}\Big(w^*_{\mathsf{inv}}\cdot z\Big)\Big(w^*_{\mathsf{inv}}\cdot z\Big)$. 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 $\mathsf{Inv\text{-}Margin} > 0$, then the labelling hyperplane w^*_{inv} separates the support into two halves (see Figure 2.1b)).

Assumption 7. ***Strictly separable invariant features.*** $\mathsf{Inv\text{-}Margin} > 0$.

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

Theorem 2. ***Impossibility of guaranteed OOD generalization for linear classification.*** *Suppose each $e \in \mathcal{E}_{all}$ follows Assumption 2. If for all the training environments $\mathcal{E}_{tr}$, 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.* (2.2.1)*), i.e., for the output of every algorithm $\exists\ e \in \mathcal{E}_{all}$ in which the error exceeds the minimum required value q (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 2.1b), 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 2.1b)), then the algorithm fails.

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

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 [**7**], 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. (2.3.1)) 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. (2.2.1)). Each classifier can be written as $\bar{w} \cdot X^e = \bar{w} \cdot S(Z^e_{\mathsf{inv}}, Z^e_{\mathsf{spu}}) = \tilde{w}_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \tilde{w}_{\mathsf{spu}} Z^e_{\mathsf{spu}}$. If $\tilde{w}_{\mathsf{spu}} \neq 0$, then the classifier $\bar{w}$ is said to rely on spurious features.

Theorem 3. ***Sufficiency and Insufficiency of ERM and IRM.*** *Suppose each $e \in \mathcal{E}_{all}$ 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.* (2.2.1)*). 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.* (2.2.1)*). 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. (2.2.1)). Interestingly, in this case we can have classifiers that rely on the spurious features and yet solve the OOD generalization (eq. (2.2.1)). For the 2D case (eq. (2.3.1)) this case implies that the entire set $\mathcal{S}$ solves the OOD generalization (eq. (2.2.1)). 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. (2.3.1)) this implies that only the predictors that rely only on X^e_{inv} in the set $\mathcal{S}$ solve the OOD generalization (eq. (2.2.1)).

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.

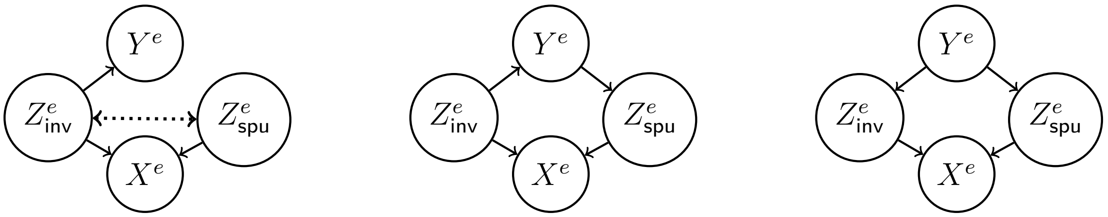
2.4. Information bottleneck principle meets invariance principle

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 [**62**]. Mutual information $I(X;\Phi(X))$ is used to measure information compression. If representation $\Phi(X)$ is a deterministic transformation of X, then in principle we can use the entropy of $\Phi(X)$ to measure compression [**32**]. Let us revisit the 2D case (eq. (2.3.1)) 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 Φ: identity (selects both features), selects invariant feature only, selects spurious feature only. The entropy of $H(\Phi(X^e))$ when Φ is the identity is $H(p^e)+\log(2)$, where $H(p^e)$ is the Shannon entropy in $\mathsf{Bernoulli}(p^e)$. If Φ selects the invariant/spurious features only, then $H(\Phi(X^e)) = \log(2)$. 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 $e \in \mathcal{E}_{tr}$, $Y^e \leftarrow X_{\mathsf{inv}}^{1,e} \oplus X_{\mathsf{inv}}^{2,e} \oplus N^e$ and $X_{\mathsf{spu}}^e \leftarrow Y^e \oplus V^e$, where all the random variables involved are binary valued, noise N^e, V^e are Bernoulli with parameters q (identical across $\mathcal{E}_{tr}$), c^e (varies across $\mathcal{E}_{tr}$) respectively. If $c^e < q$, then in $\mathcal{E}_{tr}$ predictions based on X_{spu}^e are better than predictions based on $X_{\mathsf{inv}}^{1,e}$,$X_{\mathsf{inv}}^{2,e}$. If both $X_{\mathsf{inv}}^{1,e}, X_{\mathsf{inv}}^{2,e}$ are uniform Bernoulli, then these features have a higher entropy than X_{spu}^e. In this case, the information bottleneck would bar using $X_{\mathsf{inv}}^{1,e}, X_{\mathsf{inv}}^{2,e}$. Instead, we want the model to focus on $X_{\mathsf{inv}}^{1,e}$, $X_{\mathsf{inv}}^{2,e}$ and not on X_{spu}^e. Invariance constraints encourage the model to focus on $X_{\mathsf{inv}}^{1,e}$, $X_{\mathsf{inv}}^{2,e}$. In this example, observe that invariant features are partially informative unlike the 2D case (eq. (2.3.1)).

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. (2.3.1)) and the example we highlighted in the paragraph above: $Y^e \leftarrow X_{\mathsf{inv}}^e \oplus N^e$, $X_{\mathsf{spu}}^{1,e} \leftarrow X_{\mathsf{inv}}^e \oplus W^e$, and $X_{\mathsf{spu}}^{2,e} \leftarrow Y^e \oplus V^e$. In this case, the invariance constraint does not allow representations that use $X_{\mathsf{spu}}^{2,e}$ but does not prohibit representations that rely on $X_{\mathsf{spu}}^{1,e}$. However, information bottleneck constraints on top ensure that representations that only use X_{inv}^e are used. We now describe an objective [8] that combines both these principles:

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



(a) FIIF (this work) (b) PIIF [**7**] (c) PIIF [**55**]

Fig. 2.2. Comparison of the DAG from Assumption 2 (fully informative invariant features) vs. DAGs from [**55, 7**] (partially informative invariant features).

$$\min_{w,\Phi} \sum_{e\in\mathcal{E}_{tr}} h^e\Big(w\cdot\Phi\Big) \quad \text{s.t.}\ \frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}} R^e\Big(w\cdot\Phi\Big) \leq r^{\mathsf{th}},\, w\in \arg\min_{\tilde{w}\in\mathbb{R}^{k\times r}} R^e(\tilde{w}\cdot\Phi), \forall e\in\mathcal{E}_{tr}, \quad (2.4.1)$$

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, $h(X) = -\mathbb{E}_X[\log d\mathbb{P}_X]$ 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 ζ [**32**] 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***

- ***Fully informative invariant features (FIIF).*** *Suppose each $e\in\mathcal{E}_{all}$ 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 $e\in\mathcal{E}_{tr}$ $Z^e_{\mathsf{spu}} \leftarrow AZ^e_{\mathsf{inv}} + W^e$, 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.* (2.4.1)*, with ℓ as* 0-1 *loss, and $r^{\mathsf{th}} = q$), and IB-ERM solves the OOD generalization (eq.* (2.2.1)*) but ERM and IRM (eq.*(2.2.3)*) fail.*

- ***Partially informative invariant features (PIIF).*** *Suppose each $e\in\mathcal{E}_{all}$ follows Assumption 1 and $\exists$ $e\in\mathcal{E}_{tr}$ such that $\mathbb{E}[\epsilon^e Z^e_{\mathsf{spu}}] \neq 0$. If $|\mathcal{E}_{tr}| > 2d$ and the set $\mathcal{E}_{tr}$ lies in a linear general position (a mild condition defined in the Appendix), then each solution to IB-IRM (eq.* (2.4.1)*, with ℓ as square loss, $\sigma^2_\epsilon < r^{\mathsf{th}} \leq \sigma^2_Y$, where σ^2_Y and σ^2_ϵ are the variance* *in the label and noise across $\mathcal{E}_{tr}$) and IRM (eq.(2.2.3)) solves OOD generalization (eq. (2.2.1)) 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 (2.4.1), 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 [**55**], which focused on linear classification SEMs (DAG shown in Figure 2.2c) with partially informative invariant features. Under the assumption of complete support overlap for spurious and invariant features, authors showed IRM succeeds.

2.4.1. Proposed approach

We take the three terms from the optimization in equation (2.4.1) and create a weighted combination as

$$\sum_e \Big(R^e(\Phi)+\lambda\|\nabla_{w,w=1.0}R^e(w\cdot\Phi)\|^2+\nu h^e(\Phi)\Big) \leq \sum_e \Big(R^e(\Phi)+\lambda\|\nabla_{w,w=1.0}R^e(w\cdot\Phi)\|^2+\nu h(\Phi)\Big).$$

In the LHS above, the first term corresponds to the risks across environments, the second term approximates invariance constraint (follows the IRMv1 objective [**7**]), and the third term is the entropy of the classifier in each environment. In the RHS, $h(\Phi)$ is the entropy of Φ 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 [**32**]. 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 Φ instead of

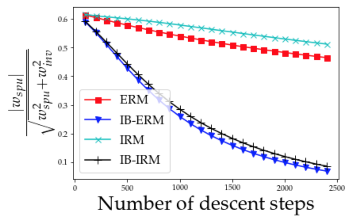


Fig. 2.3. Comparing convergence of $\frac{|w_{\mathsf{spu}}|}{\sqrt{w_{\mathsf{spu}}^2+w_{\mathsf{inv}}^2}}$ (metric from [**42**]) for average selection bias $p = 0.9$.

the differential entropy (For further details, see the Appendix). Our final objective is given as

$$\sum_{e}\Big(R^e(\Phi) + \lambda\|\nabla_{w,w=1.0}R^e(w\cdot\Phi)\|^2 + \gamma\mathsf{Var}(\Phi)\Big). \tag{2.4.2}$$

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. (2.2.1)). In some simple cases, such as the 2D case (eq. (2.3.1)), it can be shown that gradient descent is biased towards selecting the ideal classifier [**59, 42**]. 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 [**42**]. In the next theorem, we characterize the impact of using IB penalty ($\mathsf{Var}(\Phi)$) in the 2D example (eq. (2.3.1)). We compare the methods in terms of $|\frac{w_{\mathsf{spu}}(t)}{w_{\mathsf{inv}}(t)}|$, which was the metric used in [**42**]; $w_{\mathsf{spu}}(t)$ and $w_{\mathsf{inv}}(t)$ 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 $e \in \mathcal{E}_{tr}$ follows the 2D case from equation* (2.3.1)*. Set $\lambda = 0$, $\gamma > 0$ in equation* (2.4.2) *to get the IB-ERM objective with ℓ as exponential loss. Continuous-time gradient descent on this IB-ERM objective achieves $|\frac{w_{\mathsf{spu}}(t)}{w_{\mathsf{inv}}(t)}| \leq \epsilon$ in time less than $\frac{W_0(\frac{1}{2\gamma})}{2(1-p)\epsilon}$ ($W_0(\cdot)$ denotes the principal branch of the Lambert W function), while in the same time the ratio for ERM $|\frac{w_{\mathsf{spu}}(t)}{w_{\mathsf{inv}}(t)}| \geq \ln(\frac{1+2p}{3-2p})/\ln\Big(1 + \frac{W_0(\frac{1}{2\gamma})}{2(1-p)\epsilon}\Big)$, where $p = \frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}} p^e$.*

$|\frac{w_{\mathsf{spu}}(t)}{w_{\mathsf{inv}}(t)}|$ converges to zero for both methods, but it converges much faster for IB-ERM (for $p = 0.9, \epsilon = 0.001, \gamma = 0.58$, the ratio for IB-ERM is $|\frac{w_{\mathsf{spu}}(t)}{w_{\mathsf{inv}}(t)}| \leq 0.001$ and ratio for ERM is $|\frac{w_{\mathsf{spu}}(t)}{w_{\mathsf{inv}}(t)}| \geq 0.09$). 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. (2.3.1)). See Figure 3 for the comparisons.

2.5. 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 2.2, Terra Incognita dataset [**12**], and COCO [**1**]. We follow the same protocol for tuning hyperparameters from [**10, 7**] 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 2.3, we provide a comparison of methods for different examples in linear unit tests [**10**] for three and six training environments. In Table 2.4, 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 [**10**]).

2.6. 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 $X \leftarrow S(Z_{\mathsf{inv}}, Z_{\mathsf{spu}})$ in Assumption 2. Suppose S is non-linear and bijective. We can divide the learning task into two parts a) invert S to obtain $Z_{\mathsf{inv}}, Z_{\mathsf{spu}}$ and b) learn a linear model that only relies on the invariant features Z_{inv} to predict the label Y. For 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 [**31**]. The current state-of-the-art to solve part a) requires strong structural assumptions on the dependence between all the components of $Z_{\mathsf{inv}}, Z_{\mathsf{spu}}$ [**37**]. Therefore, solving part a) and part b) in conjunction with minimal assumptions forms an exciting future work. In the

[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 2.3. 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 2.4. Classification accuracy percentage on colored MNISTs, Terra Incognita and COCO dataset.

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 (2.4.2)) described in [**32**]. The approximation (even though an upper bound) serves as an effective proxy for the true information bottleneck as shown in the experiments in [**32**] (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 [**55, 29, 25**], 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 [**33, 3, 16**]. 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.

2.7. 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. We propose an approach that combines both these principles and demonstrate its effectiveness on linear unit tests [**10**] and on various high-dimensional real datasets.

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.

2.8. Appendix

Organization. In Section 2.8.1, we discuss the societal impact of this work. In Section 2.8.2, we provide further details on the experiments. In Section 1.1, we provide a detailed discussion on structural equation models and the linear general position assumption used to prove Theorem 1. In Section 2.8.4, 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 2.8.5, we provide the proof for sufficiency and insufficiency characterization of ERM and IRM discussed in Theorem 3. In Section 2.8.6, we provide the proof for Theorem 4, which compares IB-IRM, IB-ERM with IRM and ERM. In Section 2.8.7, we discuss the step-by-step derivation of the final objective in equation (2.4.2). In Section 2.8.8, we provide the proof for Theorem 5, which compares the impact of information bottleneck penalty on the learning speed. In Section 2.8.9, 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 1.3, we provide a more detailed discussion on other related works.

2.8.1. 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.

2.8.2. 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`).

2.8.2.1. Datasets. We first describe the datasets (Example 1/1S, Example 2/2S, Example 3/3S) introduced in [**10**]; these datasets are referred to as the linear unit tests. The results for linear unit tests are presented in Table 2.3.

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

$$\begin{aligned} &Z^e_{\textsf{inv}} \sim \mathcal{N}_m(0, (\sigma^e)^2), && \tilde{Y}^e \sim \mathcal{N}_m(W_{yz} Z^e_{\textsf{inv}}, (\sigma^e)^2), \\ &Z^e_{\textsf{spu}} \sim \mathcal{N}_o(W_{zy}\tilde{Y}^e, 1), && Z^e \leftarrow (Z^e_{\textsf{inv}}, Z^e_{\textsf{spu}}), \\ &Y^e \leftarrow \frac{2}{(m+o)}\mathbf{1}_m^{\mathsf{T}}\tilde{Y}^e, && X^e \leftarrow S(Z^e), \end{aligned}$$

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, $\mathbf{1}_m \in \mathbb{R}^m$ is a vector of ones, $\mathcal{N}_k$ 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 $(\sigma^{e_j})^2 \sim \textsf{Uniform}(10^{-2}, 10)$. 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 (2.3.1). Let

$$\begin{aligned} &\theta_{\textsf{cow}} = \mathbf{1}_{\textsf{m}}, && \theta_{\textsf{camel}} = -\theta_{\textsf{cow}}, && \nu_{\textsf{animal}} = 10^{-2}, \\ &\theta_{\textsf{grass}} = \mathbf{1}_{\textsf{o}}, && \theta_{\textsf{sand}} = -\theta_{\textsf{grass}}, && \nu_{\textsf{background}} = 1. \end{aligned}$$

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

$$\begin{aligned} &U^e \sim \textsf{Categorical}\Big(p^e s^e, (1-p^e)s^e, p^e(1-s^e), (1-p^e)(1-s^e)\Big), \\ &Z^e_{\textsf{inv}} \sim \begin{cases} (\mathcal{N}_m(0, 0.1) + \theta_{\textsf{cow}})\nu_{\textsf{animal}} & \text{if } U^e \in \{1,2\}, \\ (\mathcal{N}_m(0, 0.1) + \theta_{\textsf{camel}})\nu_{\textsf{animal}} & \text{if } U^e \in \{3,4\}, \end{cases} \\ &Z^e_{\textsf{spu}} \sim \begin{cases} (\mathcal{N}_o(0, 0.1) + \theta_{\textsf{grass}})\nu_{\textsf{background}} & \text{if } U^e \in \{1,4\}, \\ (\mathcal{N}_o(0, 0.1) + \theta_{\textsf{sand}})\nu_{\textsf{background}} & \text{if } U^e \in \{2,3\}, \end{cases} \\ &Z^e \leftarrow (Z^e_{\textsf{inv}}, Z^e_{\textsf{spu}}), \quad X^e \leftarrow S(Z^e), \\ &Y^e \leftarrow \textsf{I}(\mathbf{1}_m^{\mathsf{T}} Z^e_{\textsf{inv}}), \end{aligned}$$

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 $p^{e_j} \sim \textsf{Uniform}(0.9,1)$, and $s^{e_j} \sim \textsf{Uniform}(0.3, 0.7)$. 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 [**55**]. The example is meant to present a linear version of the spiral classification problem in [**44**]. Let $\theta_{\mathsf{inv}} = 0.1 \cdot \mathbf{1}_m$, and $\theta^e_{\mathsf{spu}} \sim \mathcal{N}_o(0,1)$ for all the environments. The dataset in environment $e \in \mathcal{E}_{all}$ is sampled from the following distribution

$$\begin{aligned} &Y^e \sim \mathsf{Bernoulli}\Big(\frac{1}{2}\Big), \\ &Z^e_{\mathsf{inv}} \sim \begin{cases} \mathcal{N}_m(+\theta_{\mathsf{inv}}, 0.1) \text{ if } Y^e = 0, \\ \mathcal{N}_m(-\theta_{\mathsf{inv}}, 0.1) \text{ if } Y^e = 1, \end{cases} \\ &Z^e_{\mathsf{spu}} \sim \begin{cases} \mathcal{N}_o(+\theta^e_{\mathsf{spu}}, 0.1) \text{ if } Y^e = 0, \\ \mathcal{N}_o(-\theta^e_{\mathsf{spu}}, 0.1) \text{ if } Y^e = 1, \end{cases} \qquad , \\ &Z^e \leftarrow (Z^e_{\mathsf{inv}}, Z^e_{\mathsf{spu}}), \quad X^e \leftarrow S(Z^e). \end{aligned} \tag{2.8.1}$$

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 [**7**]. 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 $Z^e_{\mathsf{spu}} = 1$, 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 [**4**], 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

[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].

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 [**12**] as formulated in [**25**]. We set up a ten-class classification task for $3 \times 224 \times 224$ 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 [**1**] (See the details in Appendix A.2 of [**1**]). 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.

2.8.2.2. Training and evaluation procedure. **Example 1/1S, 2/2S, 3/3S.** We follow the same protocol as was prescribed in [**10**] 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 γ associated with the $\mathsf{Var}(\Phi)$ term in the final objective in equation (2.4.2). 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 [**10**]. We sample the γ from $1 - 10^{\mathsf{Uniform}(-2,0)}$ following the practice in unit test experiments [**10**]. Note that the hyperparameters are trained using training environment distribution data, which is called the train-domain validation set evaluation procedure in [**25**]. 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 $1\mathrm{e} - 3$. These layers are followed by the output layer of size two. We use Adam optimizer for training with a learning rate set to $1\mathrm{e}-3$. 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, λ for IRMv1 penalty, and γ for the IB penalty. For IRM, we need to select the IRMv1 penalty λ, we set a grid of 25 values uniformly spaced in the interval $[1\mathrm{e}-1, 1.8\mathrm{e}4]$. For IB-ERM, we need to select the IB penalty γ, we set a grid of 25 values uniformly spaced in the interval $[1\mathrm{e}-1, 1.8\mathrm{e}4]$. For IB-IRM, we need to select both λ and γ, we set a 5×5 uniform grid that searches over $[1\mathrm{e}-1, 1.8\mathrm{e}4] \times [1\mathrm{e}-1, 1.8\mathrm{e}4]$. 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 [**25**] 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 [**25**], which takes samples from the same distribution as test domain and does limited model queries (we set 25 queries). In [**7**], 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 2.4 (5 percent of the total data 50000 follows the test environment distribution). In Section 2.8.2.3, we show the results for 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 $1\mathrm{e}-1$ 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, λ for IRMv1 penalty, and γ for the IB penalty. For IRM, we need to select the IRMv1 penalty λ, we set a grid of 25 values uniformly spaced in the interval $[1\mathrm{e}-1, 1.8\mathrm{e}4]$. For IB-ERM, we need to select the IB penalty γ, we set a grid of 25 values uniformly spaced in the interval $[1\mathrm{e}-1, 1.8\mathrm{e}4]$. For IB-IRM, we need to select both λ and γ, we set a 5×5 uniform grid that searches over $[1\mathrm{e}-1, 1.8\mathrm{e}4] \times [1\mathrm{e}-1, 1.8\mathrm{e}4]$. 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 2.8.2.3. 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 [**25**]. The distribution of the hyper parameters are shown in Table 2.5. Results shown in Table 2.4 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 2.5. Hyperparameters distributions for random search given included penalty of the algorithm.

Penalty	Parameter	Random distribution
All	dropout learning rate batch size weight decay	$\text{RandomChoice}([0, 0.1, 0.5])$ $10^{\text{Uniform}(-5,-3.5)}$ $2^{\text{Uniform}(3,5.5)}$ $10^{\text{Uniform}(-6,-2)}$
IRMv1	penalty weight annealing steps	$10^{\text{Uniform}(-1,5)}$ $10^{\text{Uniform}(0,4)}$
IB	penalty weight annealing steps	$10^{\text{Uniform}(-1,5)}$ $10^{\text{Uniform}(0,4)}$

COCO dataset. Other than the IB penalty, we use the exact same hyperparameters (default values) and setup as describe in Appendix B.2 of [**1**] paper and the codebase that [**1**] 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.

2.8.2.3. 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 [**25**]. 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 [**25**]. 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 2.6. AC-CMNIST. Comparisons of the methods using the train-domain validation set tuning procedure [**25**]. 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 2.7. CS-CMNIST. Comparisons of the methods using the oracle test-domain validation set tuning procedure [**25**]. The percentages in the columns indicate what fraction of the total data (50000 points) is used for validation.

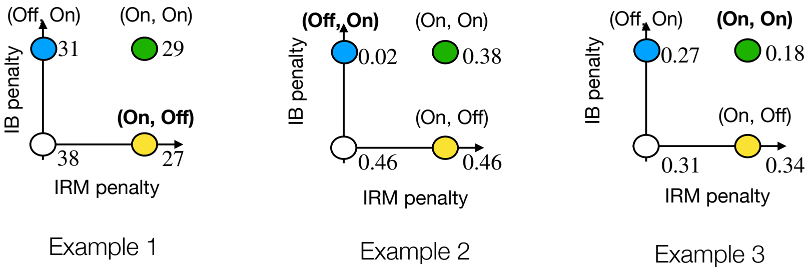
AC-CMNIST. In the CS-CMNIST dataset, for completeness, we report the accuracy of the Oracle model, which achieved a test accuracy of 99.03 ± 0.08 percent. In Table 7, we provide the supplementary experiments for CS-CMNIST carried out with train-domain validation set tuning procedure [**25**]. In Table 8, we provide the supplementary experiments for CS-CMNIST carried out with test-domain validation set tuning procedure [**25**]. 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 2.8. CS-CMNIST. Comparisons of the methods using the train-domain validation set tuning procedure [**25**]. 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 2.9. CS-CMNIST. Comparisons of the methods using the oracle test-domain validation set tuning procedure [**25**]. The percentages in the columns indicate what fraction of the total data (50000 points) is used for validation



Example 1

Example 2

Example 3

Fig. 2.4. Illustrating the impact of the IB and IRM penalty on linear unit tests [**10**]

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 [**25**]. In this section, we carry out an ablation analysis on linear unit tests [**10**] to understand the role of the different penalties. In Figure 2.4, 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.

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

2.8.2.5. 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 [**25**]

[12] `https://github.com/facebookresearch/InvariantRiskMinimization` based on [**7**]

[13] `https://github.com/facebookresearch/InvarianceUnitTests` based on [**10**]

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

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

In Theorem 1 that we informally stated from [**7**], 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 $r \in \mathbb{N}$ and for all non-zero $x \in \mathbb{R}^d$*

$$\mathsf{dim}\left(\mathsf{span}\left(\left\{\mathbb{E}_{X^e}[X^e X^{e\mathsf{T}}]x - \mathbb{E}_{X^e\epsilon^e}[X^e\epsilon^e]\right\}_{e\in\mathcal{E}_{tr}}\right)\right) > d - r. \tag{2.8.2}$$

The above assumption merely requires non-co-linearity of the training environments only. The set of matrices $\mathbb{E}_{X^e}[X^e X^{e\mathsf{T}}]$ not satisfying this assumption have a zero measure (Theorem 10 [**7**]). Consider the case when S is identity and observe that the above assumption translates to only a restriction on co-linearity of $\mathbb{E}_{Z^e}[Z^e Z^{e\mathsf{T}}]$, where $Z^e = (Z^e_{\mathsf{inv}}, Z^e_{\mathsf{spu}})$. Assume that $\mathbb{E}_{Z^e}[Z^e Z^{e\mathsf{T}}]$ 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 $\mathbb{E}_{Z^e}[Z^e Z^{e\mathsf{T}}]$ and $\mathbb{E}_{Z^e}[Z^e\epsilon^e]$ 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 $\tilde{Z}^e = \frac{Z^e}{c}$ and the matrices corresponding to it also satisfy the Assumption 8, where $c = \sqrt{\|\mathbb{E}_{Z^e}[Z^e Z^{e\mathsf{T}}]\|}$.

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

$$\begin{gathered}\mathsf{dim}\left(\mathsf{span}\left(\left\{\mathbb{E}_{Z^e}[Z^e Z^{e\mathsf{T}}]z - \mathbb{E}_{Z^e\epsilon^e}[Z^e\epsilon^e]\right\}_{e\in\mathcal{E}_{tr}}\right)\right) > d - r \implies \\ \mathsf{dim}\left(\mathsf{span}\left(\left\{\mathbb{E}_{\tilde{Z}^e}[\tilde{Z}^e \tilde{Z}^{e\mathsf{T}}]\tilde{z} - \mathbb{E}_{\tilde{Z}^e\epsilon^e}[\tilde{Z}^e\epsilon^e]\right\}_{e\in\mathcal{E}_{tr}}\right)\right) > d - r,\end{gathered} \tag{2.8.3}$$

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

$$\mathbb{E}[\bar{Z}^e \bar{Z}^{e,\mathsf{T}}] = BB^t. \tag{2.8.4}$$

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

$$B\mathbb{E}[K^e\epsilon^e] = \tilde{\rho}^e \implies \mathbb{E}[K^e\epsilon^e] = B^{-1}\tilde{\rho}^e \tag{2.8.5}$$

Thus we can conclude without loss of generality that from any set of matrices $\{\Sigma^e, \rho^e\}_{e\in\mathcal{E}_{tr}}$ 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 (2.2.3)) over such training environments with bounded support, we can still recover the ideal invariant predictor that solves the OOD generalization problem in equation (2.2.1) (i.e., $\nexists e \in \mathcal{E}_{all}$ for which risk $> \sigma^2_{\mathsf{sup}}$). 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 $\mathcal{E}_{all} \setminus \mathcal{E}_{tr}$ 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.

2.8.4. 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. $\circ$ is used to denote the composition of functions, $\cdot$ is used for matrix multiplication. $\mathbb{P}^e$ 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 $(Z^e_{\mathsf{inv}}, Z^e_{\mathsf{spu}})$. S is the matrix relating X^e and Z^e and $X^e = S(Z^e)$. w denotes a linear classifier, Φ 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 ℓ as 0-1 loss for classification, and square loss for regression. For a discrete random variable $X \in \mathbb{R}^d$, the support is defined as $\mathcal{X} = \{x \in \mathbb{R}^d \mid \mathbb{P}_X(x) > 0\}$, where $\mathbb{P}_X(x)$ is the probability of $X = x$. For a continuous random variable $X \in \mathbb{R}^d$, the support is defined as $\mathcal{X} = \{x \in \mathbb{R}^d \mid d\mathbb{P}_X(x) > 0\}$, where $d\mathbb{P}_X(x)$ is the Radon-Nikodym derivative of $\mathbb{P}_X$ w.r.t the Lebesgue measure over the completion of the Borel sets in $\mathbb{R}^d$ [**9**]. $\mathcal{Z}^e$, $\mathcal{Z}^e_{\mathsf{inv}}$, $\mathcal{Z}^e_{\mathsf{spu}}$, and $\mathcal{X}^e$ 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}. $\mathcal{X} = \mathbb{R}^d$ and $\mathcal{Y} = \{0,1\}$ for classification and $\mathcal{Y} = \mathbb{R}$ for regression. Before we can prove Theorem 2, we need to prove intermediate lemmas needed as preliminary results for it.

Define

$$\mathcal{W}_{\mathsf{inv}} = \Big\{(w_{\mathsf{inv}},0) \in \mathbb{R}^{m+o} \;\Big|\; \|w_{\mathsf{inv}}\| = 1,\ \forall z_{\mathsf{inv}} \in \cup_{e\in\mathcal{E}_{tr}} \mathcal{Z}^e_{\mathsf{inv}},\ \mathsf{I}\Big(w^*_{\mathsf{inv}}\cdot z_{\mathsf{inv}}\Big) = \mathsf{I}\Big(w_{\mathsf{inv}}\cdot z_{\mathsf{inv}}\Big)\Big\} \quad (2.8.6)$$

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

$$g^* = \mathsf{I} \circ \Big(\Big(w^*_{\mathsf{inv}},0\Big) \circ S^{-1}\Big) \quad (2.8.7)$$

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

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* (2.2.1) *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 $w^*_{\mathsf{inv}}\cdot Z^e_{\mathsf{inv}} \geq 0$, then $\mathbb{P}(Y^e = 0|X^e) < \mathbb{P}(Y^e = 1|X^e)$ (since $q < \frac{1}{2}$), which implies the prediction is 1. If

$w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} < 0$, then $\mathbb{P}(Y^e = 1|X^e) < \mathbb{P}(Y^e = 0|X^e)$, which implies the prediction is 0. We show that g^* achieves an error of q in each environment,

$$\begin{aligned} R^e(g^*) &= \mathbb{E}\Big[Y^e \oplus \mathsf{I}(w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}})\Big] \\ &= \mathbb{E}\Big[\Big(\mathsf{I}(w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}}) \oplus N^e\Big) \oplus \mathsf{I}(w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}})\Big] = q. \end{aligned} \tag{2.8.8}$$

Define $\mathcal{F}$ to be the set of all the maps $\mathbb{R}^d \rightarrow \mathcal{Y}$. From the equation (2.8.8) we get,

$$\begin{aligned} &\forall e \in \mathcal{E}_{all}, \forall f \in \mathcal{F}, R^e(f) \geq q, \\ \implies &\forall f \in \mathcal{F}, \max_{e\in\mathcal{E}_{all}} R^e(f) \geq q, \\ \implies &\min_{f\in\mathcal{F}} \max_{e\in\mathcal{E}_{all}} R^e(f) \geq q. \end{aligned} \tag{2.8.9}$$

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

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* $e \in \mathcal{E}_{all}$,

$$\begin{aligned} &Y^e \leftarrow \mathsf{I}\Big(w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}}\Big) \oplus N^e, \qquad N^e \sim \mathsf{Bernoulli}(q), q < \frac{1}{2}, \qquad N^e \perp Z^e_{\mathsf{inv}}, \\ &X^e \leftarrow S\Big(Z^e_{\mathsf{inv}}, Z^e_{\mathsf{spu}}\Big). \end{aligned} \tag{2.8.10}$$

Observe that the SEM above in Assumption 9 is analogous the the SEM in Assumption 1. Also, observe that in the above SEM $\exists\ e$ such that $N^e \not\perp Z^e_{\mathsf{spu}}$, 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* (2.2.1) *and achieves a risk of* q *in each environment.*

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

$$\begin{aligned} \forall f \in \mathcal{F}, R^{e'}(f) \geq q \implies &\forall f \in \mathcal{F}, \max_{e\in\mathcal{E}_{all}} R^e(f) \geq q, \\ \implies &\min_{f\in\mathcal{F}} \max_{e\in\mathcal{E}_{all}} R^e(f) \geq q \end{aligned} \tag{2.8.11}$$

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

Lemma 3. *If Assumption 2, 3, and 7 hold, and* $m \geq 2$, *then the set* $\mathcal{W}_{\mathsf{inv}}$ *(eq.* (2.8.6)*) consists of infinitely many hyperplanes that are not aligned with* w^*_{inv}.

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

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

$$y^*\Big(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big) \geq c. \tag{2.8.12}$$

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

$$y^*\Big(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}} + \gamma \cdot z_{\mathsf{inv}}\Big). \tag{2.8.13}$$

Using Cauchy-Schwarz inequality we get

$$|y^*(\gamma \cdot z_{\mathsf{inv}})| = |\gamma \cdot z_{\mathsf{inv}}| \leq \|\gamma\|\|z_{\mathsf{inv}}\|. \tag{2.8.14}$$

Since the support of the invariant features in training set $\cup_{e\in\mathcal{E}_{tr}}\mathcal{Z}^e_{\mathsf{inv}}$ is bounded, we set the magnitude of γ sufficiently small to control $y^*\Big(\gamma \cdot z_{\mathsf{inv}}\Big)$. Since $\cup_{e\in\mathcal{E}_{tr}}\mathcal{Z}^e_{\mathsf{inv}}$, is bounded $\exists\ z^{\mathsf{sup}} > 0$ such that $\forall z_{\mathsf{inv}} \in \cup_{e\in\mathcal{E}_{tr}}\mathcal{Z}^e_{\mathsf{inv}}$, $\|z_{\mathsf{inv}}\| < z_{\mathsf{sup}}$. If $\|\gamma\| \leq \frac{c}{2z^{\mathsf{sup}}}$, then from equation (2.8.14), we get that for each $z_{\mathsf{inv}} \in \cup_{e\in\mathcal{E}_{tr}}\mathcal{Z}^e_{\mathsf{inv}}$, $|y\Big(\gamma \cdot z_{\mathsf{inv}}\Big)| \leq \frac{c}{2}$. Using this we get for each $z_{\mathsf{inv}} \in \cup_{e\in\mathcal{E}_{tr}}\mathcal{Z}^e_{\mathsf{inv}}$

$$y^*\Big(\Big(w^*_{\mathsf{inv}} + \gamma\Big) \cdot z_{\mathsf{inv}}\Big) = y^*\Big(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big) + y^*\Big(\gamma \cdot z_{\mathsf{inv}}\Big) \geq y^* w_{\mathsf{inv}} \cdot z_{\mathsf{inv}} - |y^*\gamma \cdot z_{\mathsf{inv}}| \geq \frac{c}{2}. \tag{2.8.15}$$

From equation (2.8.12) and (2.8.15), we have that

$$\mathsf{sgn}\Big((w^*_{\mathsf{inv}} + \gamma) \cdot z_{\mathsf{inv}}\Big) = \mathsf{sgn}\Big(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big) \implies \mathsf{I}\Big((w^*_{\mathsf{inv}} + \gamma) \cdot z_{\mathsf{inv}}\Big) = \mathsf{I}\Big(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big).$$

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

$$\Big(\frac{1}{\|w^*_{\mathsf{inv}} + \gamma\|}(w^*_{\mathsf{inv}} + \gamma),0\Big) \in \mathcal{W}_{\mathsf{inv}}.$$

Also, observe that we can construct infinite such vectors that belong to $\mathcal{W}_{\mathsf{inv}}$. A simple way to check this this is consider $\gamma^{'} = \theta\gamma$, where $\theta \in (0,1)$. The same condition in equation (2.8.15) also holds with γ replaced with $\gamma^{'}$. We define this set as follows

$$\mathcal{W}_{\mathsf{inv}}(\gamma) = \Big\{\Big(\frac{1}{\|w^*_{\mathsf{inv}} + \theta\gamma\|}(w^*_{\mathsf{inv}} + \theta\gamma),0\Big) \in \mathbb{R}^{m+o}\ \Big|\ \theta \in [0,1]\Big\}, \tag{2.8.16}$$

and from the reasoning presented above it follows that $\mathcal{W}_{\mathsf{inv}}(\gamma) \subseteq \mathcal{W}_{\mathsf{inv}}$. This completes the proof.

□

We restate Theorem 2 for convenience.

Theorem 6. ***Impossibility of guaranteed OOD generalization for linear classification.*** *Suppose each $e \in \mathcal{E}_{all}$ follows Assumption 2. If for all the training environments $\mathcal{E}_{tr}$, 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.* (2.2.1)*), i.e.,* *for the output of every algorithm $\exists\, e \in \mathcal{E}_{all}$ 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 $F : \cup_{i=1}^{\infty}\Big(\mathcal{X}\times\mathcal{Y}\Big)^{i} \dots\ |\mathcal{E}_{tr}|$ times $\cup_{i=1}^{\infty}\Big(\mathcal{X}\times\mathcal{Y}\Big)^{i} \rightarrow \mathcal{Y}^{\mathcal{X}}$, where F takes as input data from each of the training environments and outputs a classifier, which takes as input a data point from $\mathcal{X}$ and outputs the label in $\mathcal{Y}$. For datasets $\{D^e\}_{e\in\mathcal{E}_{tr}}$ from the different training environments the output of the learner is $F\Big(\{D^e\}_{e\in\mathcal{E}_{tr}}\Big)$. For simplicity of notation, let us denote $F\Big(\{D^e\}_{e\in\mathcal{E}_{tr}}\Big)$ as f. We first show that if $f \neq g^*$, where g^* is defined in equation (2.8.7), then the learner cannot be OOD optimal. Take the point x where the $f \neq g^*$. 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 $1-q$ ($\mathbb{E}[f \oplus g^* \oplus N^e] = \mathbb{E}[1 \oplus N^e] = 1-q$). As a result, f cannot solve equation (2.2.1). This observation combined with Lemma 1 leads us to the conclusion that $f = g^*$ is necessary and sufficient to solve equation (2.2.1) when $\mathcal{E}_{all}$ follow Assumption 2.

We define a family of classifiers using $\mathcal{W}_{\mathsf{inv}}$ (from eq. (2.8.6)) as follows

$$\mathcal{W}^{\dagger}_{\mathsf{inv}} = \Big\{\mathsf{I}\circ\Big((w,0)\circ S^{-1}\Big)\ \Big|\ (w,0)\in\mathcal{W}_{\mathsf{inv}}\Big\}. \tag{2.8.17}$$

Next, we would like to show that the set $\mathcal{W}^{\dagger}_{\mathsf{inv}}$ consists of infinitely many distinct functions. Choose any $w^{'}_{\mathsf{inv}}$ such that $(w^{'}_{\mathsf{inv}},0)\in\mathcal{W}_{\mathsf{inv}}$ and $w^{'}_{\mathsf{inv}} \neq w^{*}_{\mathsf{inv}}$. Define $g^{'} = \mathsf{I}\circ\Big((w^{'}_{\mathsf{inv}},0)\circ S^{-1}\Big)$. We will next show that $g^* \neq g^{'}$, where g^* was defined in equation (2.8.7).

Define

$$\begin{bmatrix} w^{*}_{\mathsf{inv}} \\ w^{'}_{\mathsf{inv}} \end{bmatrix} z_{\mathsf{inv}} = \begin{bmatrix} 1 \\ -1 \end{bmatrix}. \tag{2.8.18}$$

There are two possibilities a) $w^{'}_{\mathsf{inv}}$ is not aligned with w^{*}_{inv} in which case the rank of the matrix in the above equation (2.8.18) is two and as a result the range space of the matrix spans all two-dimensional vectors, b) $w^{'}_{\mathsf{inv}}$ is aligned with w^{*}_{inv} but since $\|w^{'}_{\mathsf{inv}}\| = 1$, $w^{'}_{\mathsf{inv}} = -w^{*}_{\mathsf{inv}}$ in which case $z_{\mathsf{inv}} = w^{*}_{\mathsf{inv}}$ solves the above equation (2.8.18). In both the cases the equation (2.8.18) has a solution. Let the solution of the above equation (2.8.18) be $\tilde{z}_{\mathsf{inv}}$. Define $\tilde{x} = S\cdot(\tilde{z}_{\mathsf{inv}},0)$. Therefore, from equation (2.8.18) it follows that $g^*(\tilde{x}) \neq g^{'}(\tilde{x})$. See the simplification below for the justification.

$$\begin{aligned} g^*(\tilde{x}) &= \mathsf{I}\Big((w^{*}_{\mathsf{inv}},0)\cdot S^{-1}(\tilde{x})\Big) = \mathsf{I}(w^{*}_{\mathsf{inv}}\cdot\tilde{z}_{\mathsf{inv}}) = 1 \\ g^{'}(\tilde{x}) &= \mathsf{I}\Big((w^{'}_{\mathsf{inv}},0)\cdot S^{-1}(\tilde{x})\Big) = \mathsf{I}(w^{'}_{\mathsf{inv}}\cdot\tilde{z}_{\mathsf{inv}}) = 0 \end{aligned} \tag{2.8.19}$$

We showed above that $g^* \in \mathcal{W}^{\dagger}_{\mathsf{inv}}$ and $g^{'} \in \mathcal{W}^{\dagger}_{\mathsf{inv}}$ are two distinct functions. Recall in Lemma 4, we showed $\mathcal{W}_{\mathsf{inv}}$ has infinitely many distinct hyperplanes. We can select any pair of hyperplanes $\mathcal{W}_{\mathsf{inv}}$, for the corresponding functions in the set $\mathcal{W}^{\dagger}_{\mathsf{inv}}$ the condition in equation (2.8.18) continues to hold. Thus we can conclude that there are infinitely many distinct functions in $\mathcal{W}^{\dagger}_{\mathsf{inv}}$.

Recall we described above that an algorithm can successfully solve equation (2.2.1), if and only if the output $f = g^*$. Observe that the same exact training data $\{D^e\}_{e\in\mathcal{E}_{tr}}$ can be generated by any other labelling hyperplane $w^{'}_{\mathsf{inv}} \neq w^*_{\mathsf{inv}}$, where $(w^{'}_{\mathsf{inv}},0) \in \mathcal{W}_{\mathsf{inv}}$ (this follows from the definition of $\mathcal{W}_{\mathsf{inv}}$ in equation (2.8.6)). Define $g^{'} = \mathsf{I} \circ \Big((w^{'},0) \circ S^{-1}\Big)$, where $g^{'} \in \mathcal{W}^{\dagger}_{\mathsf{inv}}$. From the justification above, we know that $g^{'} \neq g$. Since $g^{'} \neq g^*$ the algorithm can only be successful on one of the two labelling hyperplanes $w^{'}_{\mathsf{inv}}$ or w^*_{inv}. In fact, since we showed that there are infinitely many possible distinct hyperplanes in $\mathcal{W}_{\mathsf{inv}}$, the algorithm can only succeed on one of them. To summarize, the algorithm fails almost everywhere on the entire set, $\mathcal{W}_{\mathsf{inv}}$, of equivalent generating models. This completes the proof. Λ

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 (2.8.18), 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 (2.8.18) construct a small ball around that region. Since the solution to equation (2.8.18) 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* $e \in \mathcal{E}_{all}$

$$\begin{aligned} Y^e &\leftarrow \arg\max(W_{\mathsf{inv}}^* \cdot Z_{\mathsf{inv}}^e) \\ X^e &\leftarrow S\Big(Z_{\mathsf{inv}}^e, Z_{\mathsf{spu}}^e\Big), \end{aligned} \tag{2.8.20}$$

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

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 2.1, 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.

2.8.5. 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 $\gamma \in \mathbb{R}^o$. We will derive a bound on the margin of $(w^*_{\mathsf{inv}}, \gamma)$. Consider a $z_{\mathsf{inv}} \in \cup_{e\in\mathcal{E}_{tr}} \mathcal{Z}^e_{\mathsf{inv}}$ and a $z_{\mathsf{spu}} \in \cup_{e\in\mathcal{E}_{tr}} \mathcal{Z}^e_{\mathsf{spu}}$. Define $y^* = \mathsf{sgn}(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}})$. The margin $(w^*_{\mathsf{inv}}, \gamma)$ at this point $(z_{\mathsf{inv}}, z_{\mathsf{spu}})$ with respect to y^* is defined as

$$y^*\Big(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big) + y^*\Big(\gamma \cdot z_{\mathsf{spu}}\Big). \tag{2.8.21}$$

Using Cauchy-Schwarz inequality, we get

$$|y^*\Big(\gamma \cdot z_{\mathsf{spu}}\Big)| = |\gamma \cdot z_{\mathsf{spu}}| \leq \|\gamma\|\|z_{\mathsf{spu}}\|. \tag{2.8.22}$$

Since the train support of spurious feature is bounded we can set the magnitude of γ sufficiently small to control $y^*\Big(\gamma \cdot z_{\mathsf{spu}}\Big)$. If $\|\gamma\| \leq \frac{c}{2z^{\mathsf{sup}}}$, then $|\gamma \cdot z_{\mathsf{spu}}| \leq \frac{c}{2}$, where z^{sup} satisfies the following condition – for each $z \in \cup_{e\in\mathcal{E}_{tr}} \mathcal{Z}^e_{\mathsf{spu}}$ and $\|z\| \leq z^{\mathsf{sup}}$. We can use this to find a bound on the margin as follows. Recall from equation (2.8.12) we have

$$y^*\Big(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big) \geq c. \tag{2.8.23}$$

We use the condition $|\gamma \cdot z_{\mathsf{spu}}| \leq \frac{c}{2}$ in the simplification below

$$y^*\Big(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big) + y^*\Big(\gamma \cdot z_{\mathsf{spu}}\Big) \geq c - |\gamma \cdot z_{\mathsf{spu}}| \geq \frac{c}{2}. \tag{2.8.24}$$

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

$$\begin{aligned} R^e(g^*_{\mathsf{spu}}) &= \mathbb{E}\Big[Y^e \oplus \mathsf{I}\Big((w^*_{\mathsf{inv}}, \gamma) \cdot (z_{\mathsf{inv}}, z_{\mathsf{spu}})\Big)\Big] \\ &= \mathbb{E}\Big[\mathsf{I}\Big((w^*_{\mathsf{inv}}, 0) \cdot (z_{\mathsf{inv}}, z_{\mathsf{spu}})\Big) \oplus N^e \oplus \mathsf{I}\Big((w^*_{\mathsf{inv}}, \gamma) \cdot (z_{\mathsf{inv}}, z_{\mathsf{spu}})\Big)\Big] = \mathbb{E}\Big[N^e\Big] = q. \end{aligned} \tag{2.8.25}$$

The same calculation as above equation (2.8.25) holds in all the training environments. Thus g^*_{spu} achieves the minimum error possible q for all the training environments $e \in \mathcal{E}_{tr}$. Λ

We restate Theorem 3 for convenience.

Theorem 7. ***Sufficiency and Insufficiency of ERM and IRM.*** *Suppose each $e \in \mathcal{E}_{all}$ 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.* (2.2.1)*). 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.* (2.2.1)*). 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 (2.2.1)). Since Assumptions 2, 4, 7, hold we can use Lemma 4. From Lemma 4, it follows that for each $z_{\mathsf{inv}} \in \cup_{e\in\mathcal{E}_{tr}}\mathcal{Z}^e_{\mathsf{inv}}$ and for each $z_{\mathsf{spu}} \in \cup_{e\in\mathcal{E}_{tr}}\mathcal{Z}^e_{\mathsf{inv}}$:

$$\mathsf{I}\Big((w^*_{\mathsf{inv}}, \gamma)\cdot(z_{\mathsf{inv}}, z_{\mathsf{spu}})\Big) = \mathsf{I}\Big((w^*_{\mathsf{inv}},0)\cdot(z_{\mathsf{inv}}, z_{\mathsf{spu}})\Big). \tag{2.8.26}$$

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

$$\mathsf{I}\Big((w^*_{\mathsf{inv}}, \gamma)\cdot(z_{\mathsf{inv}}, z_{\mathsf{spu}})\Big) = \mathsf{I}\Big((w^*_{\mathsf{inv}},0)\cdot(z_{\mathsf{inv}}, z_{\mathsf{spu}})\Big) \tag{2.8.27}$$

Therefore, the error of the classifier $g^*_{\mathsf{spu}} = \mathsf{I}\circ(w^*_{\mathsf{inv}},\gamma)\circ S^{-1}$ in each environment $e \in \mathcal{E}_{all}$ is

$$\begin{aligned}
R^e(g^*_{\mathsf{spu}}) &= \mathbb{E}\Big[Y^e \oplus \mathsf{I}\Big((w^*_{\mathsf{inv}}, \gamma)\cdot(z_{\mathsf{inv}}, z_{\mathsf{spu}})\Big)\Big] \\
&= \mathbb{E}\Big[\mathsf{I}\Big((w^*_{\mathsf{inv}},0)\cdot(z_{\mathsf{inv}}, z_{\mathsf{spu}})\Big)\oplus N^e \oplus \mathsf{I}\Big((w^*_{\mathsf{inv}}, \gamma)\cdot(z_{\mathsf{inv}}, z_{\mathsf{spu}})\Big)\Big] = \mathbb{E}\Big[N^e\Big] = q.
\end{aligned} \tag{2.8.28}$$

g^*_{spu} is Bayes optimal on each environment $e \in \mathcal{E}_{all}$. Therefore, g^*_{spu} also solves equation (2.2.1). 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 $\mathcal{E}_{tr}$ and $\mathcal{E}_{all}$, each solution to ERM and IRM continues to succeed in $\mathcal{E}_{all}$. This completes the first part of the proof.

[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.

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 $\mathsf{I} \circ w$, where I is composed with a linear function. The classifier operates on x, and we get $\mathsf{I}(w \cdot x)$ and since $x = Sz$ (from Assumption 2) we can write this as $\mathsf{I}(w \cdot S(z))$. To simplify notation, we call $\mathsf{I} \circ w \circ S = \mathsf{I} \circ \tilde{w}$. Our goal is to show that if $\tilde{w}$ assigns a non-zero weight to the spurious features, then $\mathsf{I} \circ w \circ S$ cannot solve the OOD generalization problem (eq. (2.2.1)). We write $\tilde{w} = (\tilde{w}_{\mathsf{inv}}, \tilde{w}_{\mathsf{spu}})$. Suppose $\tilde{w}_{\mathsf{spu}} \neq 0$ and yet the classifier solves the problem in equation (2.2.1). Consider the classifier that generates the data $(w^*_{\mathsf{inv}}$,0). Pick any point $z_{\mathsf{inv}} \in \cup_{e \in \mathcal{E}_{all}} \mathcal{Z}^e_{\mathsf{inv}}$ and pick any non-zero $z^e_{\mathsf{spu}} \in \mathbb{R}^o$. Call $z = (z_{\mathsf{inv}}, z_{\mathsf{spu}})$ We divide the analysis into two cases.

Case 1: $\mathsf{I}\Big((\tilde{w}_{\mathsf{inv}}, \tilde{w}_{\mathsf{spu}}) \cdot z\Big) \neq \mathsf{I}\Big((w^*_{\mathsf{inv}}, 0) \cdot z\Big)$. In this case, $(\tilde{w}_{\mathsf{inv}}, \tilde{w}_{\mathsf{spu}})$ cannot solve equation (2.2.1) as there exists a test environment where we have all the mass on z.

Case 2: $\mathsf{I}\Big((\tilde{w}_{\mathsf{inv}}, \tilde{w}_{\mathsf{spu}}) \cdot z\Big) = \mathsf{I}\Big((w^*_{\mathsf{inv}}, 0) \cdot z\Big)$. Observe that since $\tilde{w}_{\mathsf{spu}} \neq 0$, we can increase or decrease one of the components of z_{spu} corresponding to a non-zero $\tilde{w}_{\mathsf{spu}}$ 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 $w \in \mathcal{W}_{\mathsf{inv}}$ (from equation (2.8.6)) written as $w =$ $(w_{\mathsf{inv}}$,0). For such a classifier by definition it is true that for each $z_{\mathsf{inv}} \in \cup_{e \in \mathcal{E}_{tr}} \mathcal{Z}^e_{\mathsf{inv}}$, $\mathsf{I}\Big(w_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big) = \mathsf{I}\Big(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big)$. From Assumption 5 it follows that for each $z_{\mathsf{inv}} \in \cup_{e \in \mathcal{E}_{all}} \mathcal{Z}^e_{\mathsf{inv}}$, $\mathsf{I}\Big(w_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big) = \mathsf{I}\Big(w^*_{\mathsf{inv}} \cdot z_{\mathsf{inv}}\Big)$ and thus the classifier continues to achieve an error of q on all the test environments. Thus we can conclude that $\mathsf{I} \circ w \circ S^{-1}$ is OOD optimal. Therefore, all the elements in the set $\mathcal{W}^{\dagger}_{\mathsf{inv}}$ (from eq. (2.8.17)) are OOD optimal.

Λ

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 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.

2.8.6. 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 $X \sim \mathbb{P}_X$ with support $\mathcal{X}$ is defined as

$$H(X) = -\sum_{x\in\mathcal{X}} \mathbb{P}_X(X = x)\log\Big(\mathbb{P}_X(X = x)\Big). \tag{2.8.29}$$

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

$$h(X) = -\int_{x\in\mathcal{X}} \log\Big(d\mathbb{P}_X(x)\Big)d\mathbb{P}_X(x), \tag{2.8.30}$$

where $d\mathbb{P}_X(x)$ is the Radon-Nikodym derivative of $\mathbb{P}_X$ w.r.t the Lesbegue measure.

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

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

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

$$\begin{aligned}
H(Z|X) &= -\sum_{x\in\mathcal{X}} \mathbb{P}_X(x) \sum_{z\in\mathcal{Z}} \mathbb{P}_{Z|X}(Z = z|X = x)\log\Big(\mathbb{P}_{Z|X}(Z = z|X = x)\Big)\\
&= -\sum_{x\in\mathcal{X}} \mathbb{P}_X(x) \sum_{z\in\mathcal{Z}} \mathbb{P}_{Y|X}(Y = z-x|X = x)\log\Big(\mathbb{P}_{Y|X}(Y = z-x|X = x)\Big)\\
&= -\sum_{x\in\mathcal{X}} \mathbb{P}_X(x) \sum_{z\in\mathcal{Z}} \mathbb{P}_{Y|X}(Y = z-x|X = x)\log\Big(\mathbb{P}_{Y|X}(Y = z-x|X = x)\Big) \text{ (use } X \perp Y)\\
&= -\sum_{x\in\mathcal{X}} \mathbb{P}_X(x) \sum_{z\in\mathcal{Z}} \mathbb{P}_{Y}(Y = z-x)\log\Big(\mathbb{P}_{Y}(Y = z-x)\Big)\\
&= H(Y)
\end{aligned} \tag{2.8.31}$$

$$\begin{aligned}
I(Z;X) &= H(Z) - H(Z|X) = H(X+Y) - H(Y)\\
I(Z;Y) &= H(Z) - H(Z|Y) = H(X+Y) - H(X)
\end{aligned} \tag{2.8.32}$$

From equation (2.8.32) and the property of mutual information that $I(Z;X) \geq 0, I(Z;Y) \geq 0$ it follows that

$$H(X+Y) \geq H(Y),\ H(X+Y) \geq H(X) \implies H(X+Y) \geq \max\{H(X),H(Y)\}. \tag{2.8.33}$$

This completes the proof. Λ

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

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

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

$$\begin{aligned} h(Z|X) &= \mathbb{E}_{\mathbb{P}_X}\Big[\mathbb{E}_{\mathbb{P}_{Z|X}}\Big[\log\Big(d\mathbb{P}_{Z|X}(Z=z|X=x)\Big)\Big]\Big] \\ &= \mathbb{E}_{\mathbb{P}_X}\Big[\mathbb{E}_{\mathbb{P}_{Y|X}}\Big[\log\Big(d\mathbb{P}_{Y|X}(Y=z-x|X=x)\Big)\Big]\Big] \text{ (use } X \perp Y) \\ &= h(Y) \end{aligned} \tag{2.8.34}$$

Note that $I(Z;X) \geq 0 \implies h(Z) \geq h(Z|X)$. Combining this with the above equation (2.8.34) we get

$$h(X+Y) \geq h(Y). \tag{2.8.35}$$

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

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*

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

Proof of Lemma 7. Suppose $|\mathcal{X}| = \{x_{\mathsf{min}}, \dots, x_{\mathsf{max}}\}$ and $\mathcal{Y} = \{y_{\mathsf{min}}, \dots, y_{\mathsf{max}}\}$. The smallest value of $X+Y$ is $x_{\mathsf{min}} + y_{\mathsf{min}}$ and the largest value is $x_{\mathsf{max}} + y_{\mathsf{max}}$. 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 (2.8.32) it follows that $I(X+Y;Y) = 0 \implies X+Y \perp Y$. Observe that if $Z = x_{\mathsf{max}} + y_{\mathsf{max}} \implies Y = y_{\mathsf{max}}$. Therefore, $\mathbb{P}(Y = y_{\mathsf{max}}|Z = x_{\mathsf{max}} + y_{\mathsf{max}}) = 1$. However, $\mathbb{P}(Y = y_{\mathsf{max}}) \neq 1$ 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 $H(X+Y) > \max\{H(X), H(Y)\}$. Λ

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

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

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 $[x_{\mathsf{max}} - \Delta, x_{\mathsf{max}}]$. Similarly for Y, we write the rightmost interval as $[y_{\mathsf{max}} - \Delta, y_{\mathsf{max}}]$. [16] Define an event $\mathcal{M} : x_{\mathsf{max}} + y_{\mathsf{max}} - \delta \leq X + Y \leq x_{\mathsf{max}} + y_{\mathsf{max}}$. If $\mathcal{M}$ occurs, then $Y \geq y_{\mathsf{max}} - \delta$ and $X \geq x_{\mathsf{max}} - \delta$.

$$\begin{aligned}\mathbb{P}_X(X \leq x_{\mathsf{max}} - \delta|\mathcal{M}) = 0 \\ \mathbb{P}_Y(Y \leq y_{\mathsf{max}} - \delta|\mathcal{M}) = 0\end{aligned} \tag{2.8.36}$$

If $\delta < \Delta$ we know that

$$\begin{aligned}\mathbb{P}_X(X \leq x_{\mathsf{max}} - \delta) > 0 \\ \mathbb{P}_Y(Y \leq y_{\mathsf{max}} - \delta) > 0\end{aligned} \tag{2.8.37}$$

If $X + Y \perp Y$ then $\mathbb{P}_Y(Y \leq y_{\mathsf{max}} - \delta) = \mathbb{P}_Y(Y \leq y_{\mathsf{max}} - \delta|\mathcal{M})$, which is not the case from the above equations (2.8.36) and (2.8.37). 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. Λ

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.

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

$$\begin{aligned}&\min_{w\in\mathbb{R}^{k\times r},\Phi\in\mathbb{R}^{r\times d}} \frac{1}{|\mathcal{E}_{tr}|}\sum_e H^e\Big(w\cdot\Phi\Big) \\ &\text{s.t. } \frac{1}{|\mathcal{E}_{tr}|}\sum_e R^e\Big(w\cdot\Phi\Big) \leq r^* \\ &\quad w \in \arg\min_{\tilde{w}\in\mathbb{R}^{k\times r}} R^e(\tilde{w}\cdot\Phi)\end{aligned} \tag{2.8.38}$$

Note that the only difference between equation (2.8.38) and the equation (2.4.1) is that the objective here is Shannnon entropy, while the objective in the other case is the differential entropy.

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

Fully informative invariant features (FIIF). *Suppose each $e \in \mathcal{E}_{all}$ follows Assumption 2. Assume that the invariant features are strictly separable, bounded, and satisfy*

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

support overlap (Assumptions 3,5 and 7 hold). Also, for each $e \in \mathcal{E}_{tr}$ $Z^e_{\mathsf{spu}} \leftarrow AZ^e_{\mathsf{inv}} + W^e$, 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. (2.4.1)*, with ℓ as* 0-1 *loss, and $r^{\mathsf{th}} = q$), and IB-ERM solves the OOD generalization (eq.* (2.2.1)*) but ERM and IRM (eq.*(2.2.3)*) 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 ($\|Z^e_{\mathsf{spu}}\| \leq \|A\|\|Z^e_{\mathsf{inv}}\| + \|W^e\|$). 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 (2.8.38) and call it $\Phi^{\dagger}$. Consider the prediction made by this model

$$\Phi^{\dagger} \cdot X^e = \Phi^{\dagger} \cdot S(Z^e_{\mathsf{inv}}, Z^e_{\mathsf{spu}}) = \Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot Z^e_{\mathsf{spu}}. \tag{2.8.39}$$

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

$$\begin{aligned}
&\Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot Z^e_{\mathsf{spu}} \\
&= \Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot (AZ^e_{\mathsf{inv}} + W^e) \\
&= \Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot (AZ^e_{\mathsf{inv}} + W^e) \\
&= \Big[\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big] \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot W^e.
\end{aligned} \tag{2.8.40}$$

We will show that $\Phi^{+} = \Big(\Big[\Phi_{\mathsf{inv}}+\Phi_{\mathsf{spu}}\cdot A\Big], 0\Big)S^{-1} = \Big[\Phi_{\mathsf{inv}}+\Phi_{\mathsf{spu}}\cdot A\Big]S^{\dagger}_{\mathsf{inv}}$, where $S^{\dagger}_{\mathsf{inv}}$ 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 $\Phi^{\dagger}$. Recall that $\Phi^{\dagger}$ achieves an average error across the training environments of q (because $r^{\mathsf{th}} = q$ 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 $e \in \mathcal{E}_{tr}$. Since the error $\Phi^{\dagger}$ is q it implies that for each training environment e

$$\mathsf{I}(w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}}) = \mathsf{I}(\Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot Z^e_{\mathsf{spu}}) \tag{2.8.41}$$

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

$$\begin{aligned}
&R^e(\Phi^{\dagger}) = \mathbb{E}\Big[\Big(\mathsf{I}(w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}}) \oplus N^e \oplus \mathsf{I}(\Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot Z^e_{\mathsf{spu}})\Big)\Big] \\
&= \theta\mathbb{E}[1 \oplus N^e] + (1-\theta)\mathbb{E}[N^e] > q
\end{aligned} \tag{2.8.42}$$

If the above is true, then that contradicts the claim that $\Phi^{\dagger}$ achieves an error of q. Thus the statement in equation (2.8.41) has to hold at all points in the training support of the invariant features. Let $\mathcal{W}^e$ be the support of W^e. In each training environment, if we consider a $z^e_{\mathsf{inv}} \in \mathcal{Z}^e_{\mathsf{inv}}$, then $\forall w^e \in \mathcal{W}^e$, the following holds – if $\mathsf{I}(w^*_{\mathsf{inv}} \cdot z^e_{\mathsf{inv}}) = 1$, then

$$\begin{aligned}
&\Phi_{\mathsf{inv}} \cdot z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot (Az^e_{\mathsf{inv}} + w^e) \geq 0 \\
\implies &\Phi_{\mathsf{inv}} \cdot z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot (Az^e_{\mathsf{inv}}) \geq -\Phi_{\mathsf{spu}} \cdot w^e \\
\implies &\Big(\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big) \cdot z^e_{\mathsf{inv}} \geq \max_{w^e \in \tilde{\mathcal{W}}^e} -\Phi_{\mathsf{spu}} \cdot w^e \\
\implies &\Big(\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big) \cdot z^e_{\mathsf{inv}} \geq 0 \\
\implies &\Phi^{+}X^e \geq 0.
\end{aligned} \tag{2.8.43}$$

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

$$\begin{aligned} &\Big(\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big) \cdot z^e_{\mathsf{inv}} < 0 \\ &\Phi^+ X^e < 0. \end{aligned} \tag{2.8.44}$$

In the above simplification equation (2.8.43), we use $\max_{w^e} -\Phi_{\mathsf{spu}} \cdot w^e \geq 0$. Consider any component of $-\Phi_{\mathsf{spu}}$; if the sign of the component is positive (negative), then set the corresponding component of w^e to be positive (negative). As a result, $-\Phi_{\mathsf{spu}} \cdot w^e \geq 0$. In this argument, we only relied on the assumption that w^e can take both signs in the set $\mathcal{W}^e$. Suppose $\mathcal{W}^e$ 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 (2.8.43) and (2.8.44), we can conclude that Φ^+ achieves the same error of q in all the training environments.

Observe that we can write $\Phi^\dagger \cdot X^e = \Phi^+ \cdot X^e + \Phi_{\mathsf{spu}} \cdot W^e$. We state two properties that we use to show that entropy Φ^+ is smaller than $\Phi^\dagger$:

a) $\Phi_{\mathsf{spu}} \cdot W^e \perp \Phi^+ \cdot X^e$ ($\Phi^+ \cdot X^e = \Big[\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big] \cdot Z^e_{\mathsf{inv}}$ and $Z^e_{\mathsf{inv}} \perp W^e$),

b) $\Phi^+ \cdot X$, $\Phi_{\mathsf{spu}} \cdot W^e$ are discrete random variables with finite support of size at least two.

We justify why b) is true in the above. $\Phi^+ \cdot X^e$ 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). $\Phi^+ \cdot X^e$ has at least two elements in its support this follows from equation (2.8.43) and (2.8.44). $\Phi_{\mathsf{spu}} \cdot W^e$ 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 $\Phi^+ \cdot X^e$ is a classifier with lower entropy. We already established that Φ^+ achieves the same error as $\Phi^\dagger$ for all the training environments. 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 (2.8.61) is automatically satisfied. Therefore, Φ^+ is strictly preferable to $\Phi^\dagger$. Thus the solution $\Phi^\dagger$ cannot rely on the spurious features and $\Phi_{\mathsf{spu}} = 0$.

Thus any solution $\Phi^\dagger$ to equation (2.8.38) has to satisfy $\Phi^\dagger \cdot S = (\Phi_{\mathsf{inv}}, 0)$ and $\Phi^\dagger \cdot S$ also satisfies

$$\mathsf{I}(w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}}) = \mathsf{I}(\Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}}). \tag{2.8.45}$$

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 (2.8.55) 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 (2.8.38) also solves equation (2.2.1). Λ

2.8.6.2. Continuous random variables. In this section, we assume that in each $e \in \mathcal{E}_{all}$, the random variables Z^e_{inv},Z^e_{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 [**32**], we add an independent noise term to the predictor to ensure that the entropy is lower bounded. Suppose $w \cdot \Phi$ is the output of the predictor and the entropy of the predictor for the data in environment e as $h^e(w \cdot \Phi)$. Consider a prediction made by the classifier $w \cdot \Phi(X^e)$; we add noise κ^e (continuous, bounded random variable with a finite entropy) to this prediction to get $w \cdot \Phi(X) + \kappa^e$. The differential entropy after noise addition as $h^e(w \cdot \Phi(X^e) + \kappa^e)$. Observe that $h^e(w \cdot \Phi(X^e) + \kappa^e) \geq h(\kappa^e)$. In the rest of the discussion, we just write $h^e(w \cdot \Phi(X^e) + \kappa^e)$ as $h^e(w \cdot \Phi)$ to make the notation less cumbersome. We constrain $\mathcal{H}_\Phi$ ($\mathcal{H}_w$) in the optimization in equation (2.4.1) to a set $\tilde{\mathcal{H}}_\Phi = \{\Phi \in \mathbb{R}^{r \times d} \mid 0 < \phi_{\mathsf{inf}} \leq \|\Phi\| \leq \phi_{\mathsf{sup}}\}$ ($\tilde{\mathcal{H}}_w = \{w \in \mathbb{R}^{k \times r} \mid 0 < w_{\mathsf{inf}} \leq \|w\| \leq w_{\mathsf{sup}}\}$) instead of $\mathcal{H}_\Phi = \mathbb{R}^{r \times d}$ ($\mathcal{H}_w = \mathbb{R}^{k \times r}$). 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 (2.4.1) after accounting for the pathologies of differential entropy that we described above:

$$\begin{aligned} &\min_{w \in \tilde{\mathcal{H}}_w, \Phi \in \tilde{\mathcal{H}}_\Phi} \frac{1}{|\mathcal{E}_{tr}|} \sum_e h^e\Big(w \cdot \Phi\Big) \\ &\text{s.t. } \frac{1}{|\mathcal{E}_{tr}|} \sum_e R^e\Big(w \cdot \Phi\Big) \leq r^{\mathsf{th}} \\ &w \in \arg\min_{\tilde{w} \in \tilde{\mathcal{H}}_w} R^e(\tilde{w} \cdot \Phi) \end{aligned} \tag{2.8.46}$$

We restate Theorem 4 for convenience.

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

- ***Fully informative invariant features (FIIF).*** *Suppose each $e \in \mathcal{E}_{all}$ 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 $e \in \mathcal{E}_{tr}$ $Z^e_{\mathsf{spu}} \leftarrow AZ^e_{\mathsf{inv}} + W^e$, 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.* (2.4.1)*, with ℓ as* 0-1 *loss, and $r^{\mathsf{th}} = q$), and IB-ERM solves the OOD generalization (eq.* (2.2.1)*) but ERM and IRM (eq.*(2.2.3)*) fail.*

- ***Partially informative invariant features (PIIF).*** *Suppose each $e \in \mathcal{E}_{all}$ follows Assumption 1 and $\exists$ $e \in \mathcal{E}_{tr}$ such that $\mathbb{E}[\epsilon^e Z^e_{\mathsf{spu}}] \neq 0$. If $|\mathcal{E}_{tr}| > 2d$ and the set $\mathcal{E}_{tr}$ lies in a linear general position (a mild condition defined in the Appendix), then each solution to IB-IRM (eq.* (2.4.1)*, with ℓ as square loss, $\sigma^2_\epsilon < r^{\mathsf{th}} \leq \sigma^2_Y$, where σ^2_Y and σ^2_ϵ are the variance in the label and noise across $\mathcal{E}_{tr}$) and IRM (eq.*(2.2.3)*) solves OOD generalization (eq.* (2.2.1)*) 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 ($\|Z^e_{\mathsf{spu}}\| \leq \|A\|\|Z^e_{\mathsf{inv}}\| + \|W^e\|$). 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. [17]

Consider a solution to IB-IRM (eq. (2.8.46)) and call it $\Phi^{\dagger}$. Consider the prediction made by this model

$$\Phi^{\dagger} \cdot X^e = \Phi^{\dagger} \cdot S(Z^e_{\mathsf{inv}}, Z^e_{\mathsf{spu}}) = \Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot Z^e_{\mathsf{spu}}. \tag{2.8.47}$$

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

$$\begin{aligned}
&\Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot Z^e_{\mathsf{spu}} \\
&= \Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot (AZ^e_{\mathsf{inv}} + W^e) \\
&= \Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot (AZ^e_{\mathsf{inv}} + W^e) \\
&= \Big[\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big] \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot W^e.
\end{aligned} \tag{2.8.48}$$

We will show that $\Phi^{+} = \Big(\Big[\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big], 0\Big) S^{-1} = \Big[\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big] S^{\dagger}_{\mathsf{inv}}$, where $S^{\dagger}_{\mathsf{inv}}$ 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 $\Phi^{\dagger}$. Recall that $\Phi^{\dagger}$ achieves an average error across the training environments of q (because $r^{\mathsf{th}} = q$ 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 $e \in \mathcal{E}_{tr}$. Since the error $\Phi^{\dagger}$ is q it implies that for each training environment

$$\mathsf{I}(w^*_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}}) = \mathsf{I}(\Phi_{\mathsf{inv}} \cdot Z^e_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot Z^e_{\mathsf{spu}}), \tag{2.8.49}$$

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

[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.

compute the error

$$\begin{aligned} R^e(\Phi^{\dagger}) &= \mathbb{E}\Big[\Big(\mathsf{I}(w_{\mathsf{inv}}^* \cdot Z_{\mathsf{inv}}^e) \oplus N^e \oplus \mathsf{I}(\Phi_{\mathsf{inv}} \cdot Z_{\mathsf{inv}}^e + \Phi_{\mathsf{spu}} \cdot Z_{\mathsf{spu}}^e)\Big)\Big] \\ &= \theta \mathbb{E}[1 \oplus N^e] + (1-\theta)\mathbb{E}[N^e] > q \end{aligned} \tag{2.8.50}$$

If the above is true, then that contradicts the claim that $\Phi^{\dagger}$ achieves an error of q. Thus the statement in equation (2.8.49) has to hold with probability 1. Let $\mathcal{W}^e$ denote the support of W^e in environment e. We can restate the above observation as – there exists sets $\tilde{\mathcal{Z}}_{\mathsf{inv}}^e \subseteq \mathcal{Z}_{\mathsf{inv}}^e$ and a set $\tilde{\mathcal{W}}^e \subseteq \mathcal{W}^e$ such that $\mathbb{P}(\tilde{\mathcal{Z}}_{\mathsf{inv}}^e \times \tilde{\mathcal{W}}^e) = 1$ [18] and for each element in $\tilde{\mathcal{Z}}_{\mathsf{inv}}^e \times \tilde{\mathcal{W}}^e$

$$\mathsf{I}(w_{\mathsf{inv}}^* \cdot Z_{\mathsf{inv}}^e) = \mathsf{I}(\Phi_{\mathsf{inv}} \cdot Z_{\mathsf{inv}}^e + \Phi_{\mathsf{spu}} \cdot Z_{\mathsf{spu}}^e) \tag{2.8.51}$$

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

$$\begin{aligned} &\Phi_{\mathsf{inv}} \cdot z_{\mathsf{inv}}^e + \Phi_{\mathsf{spu}} \cdot (A z_{\mathsf{inv}}^e + w^e) \geq 0 \\ \implies &\Phi_{\mathsf{inv}} \cdot z_{\mathsf{inv}}^e + \Phi_{\mathsf{spu}} \cdot (A z_{\mathsf{inv}}^e) \geq -\Phi_{\mathsf{spu}} \cdot w^e \\ \implies &\Big(\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big) \cdot z_{\mathsf{inv}}^e \geq \max_{w^e \in \tilde{\mathcal{W}}^e} -\Phi_{\mathsf{spu}} \cdot w^e \\ \implies &\Big(\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big) \cdot z_{\mathsf{inv}}^e \geq 0 \\ \implies &\Phi^+ X^e \geq 0. \end{aligned} \tag{2.8.52}$$

Similarly, we can argue that if $\mathsf{I}(w_{\mathsf{inv}}^* \cdot z_{\mathsf{inv}}^e) = 0$, then

$$\begin{aligned} &\Big(\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big) \cdot z_{\mathsf{inv}}^e < 0 \\ &\Phi^+ X^e < 0. \end{aligned} \tag{2.8.53}$$

In the above simplification in equation (2.8.52), we use $\max_{w^e} -\Phi_{\mathsf{spu}} \cdot w^e \geq 0$. Consider any component of $-\Phi_{\mathsf{spu}}$; if the sign of the component is positive (negative), then set the corresponding component of w^e to be positive (negative). As a result, $-\Phi_{\mathsf{spu}} \cdot w^e \geq 0$. In this argument, we only relied on the assumption that w^e can take both signs in the set $\tilde{\mathcal{W}}^e$. Suppose w^e can only take either positive or negative values in $\tilde{\mathcal{W}}^e$ 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 (2.8.52), (2.8.53), and $\mathbb{P}(\tilde{\mathcal{Z}}_{\mathsf{inv}}^e \times \tilde{\mathcal{W}}^e) = 1$, we can conclude that Φ^+ achieves the same error of q in all the training environments.

Observe that we can write $\Phi^{\dagger} \cdot X^e = \Phi^+ \cdot X^e + \Phi_{\mathsf{spu}} \cdot W^e$. We state two properties that we use to show that entropy Φ^+ is smaller than $\Phi^{\dagger}$:

a) $\Phi_{\mathsf{spu}} \cdot W^e \perp \Phi^+ \cdot X^e$ ($\Phi^+ \cdot X^e = \Big[\Phi_{\mathsf{inv}} + \Phi_{\mathsf{spu}} \cdot A\Big] \cdot Z_{\mathsf{inv}}^e$ and $Z_{\mathsf{inv}}^e \perp W^e$),

b) $\Phi_{\mathsf{inv}}^+ \cdot X$, $\Phi_{\mathsf{spu}} \cdot W^e$ are continuous bounded random variables,

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

We justify why b) is true in the above. $\Phi_{\mathsf{inv}}^{+} \cdot X^e$ is a bounded random variable (Z_{spu}^e is bounded as Z_{inv}^e is bounded and as a result X^e is bounded as well). Observe that $\Phi_{\mathsf{inv}}^{+} \neq 0$, this follows from equation (2.8.52) and (2.8.53). $\Phi_{\mathsf{inv}}^{+} \cdot X^e$ is a continuous random variable as well. Suppose $\Phi_{\mathsf{inv}}^{+} \cdot X^e$ was not continuous, which implies for some constant b, $\Phi_{\mathsf{inv}}^{+} \cdot X^e = b$ with a finite probability. If $\Phi_{\mathsf{inv}}^{+} \cdot X^e = b$ 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

$$h^e(\Phi^{+} \cdot X^e) < h^e(\Phi^{\dagger} \cdot X^e) \tag{2.8.54}$$

Note that the above equation (2.8.54) 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 (2.8.61) 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 $\Phi_{\mathsf{spu}} = 0$

Thus any solution $\Phi^{\dagger}$ to equation (2.8.46) has to satisfy $\Phi^{\dagger} \cdot S = (\Phi_{\mathsf{inv}}, 0)$ and $\Phi^{\dagger} \cdot S$ also satisfies

$$\mathsf{I}(w_{\mathsf{inv}}^{*} \cdot Z_{\mathsf{inv}}^e) = \mathsf{I}(\Phi_{\mathsf{inv}} \cdot Z_{\mathsf{inv}}^e) \tag{2.8.55}$$

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 (2.8.55) for all the points in the support, then under the support overlap of invariant features such a solution is OOD optimal (solves equation (2.2.1)) 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 (2.8.55) 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 (2.8.46) also solves equation (2.2.1). Next, we show why it suffices for the equation (2.8.55) to hold with probability one.

Since the equation (2.8.55) 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 (2.2.1) is violated. We now argue using arguments based on standard measure theory [**9**] that such pathological shifts cannot occur under the assumptions made in this setting.

Recall that we defined $\tilde{\mathcal{Z}}_{\mathsf{inv}}^e \times \tilde{\mathcal{W}}^e$ to be the set where equation (2.8.55) holds pointwise. $\mathbb{P}(\tilde{\mathcal{Z}}_{\mathsf{inv}}^e \times \tilde{\mathcal{W}}^e) = 1$. Owing to the independence $Z^e \perp W^e$, we have $\mathbb{P}(\tilde{\mathcal{Z}}_{\mathsf{inv}}^e) = 1$, $\mathbb{P}(\tilde{\mathcal{W}}^e) = 1$. It can be shown that the Lebesgue measure μ of the set $\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e$ is zero, i.e., $\mu(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) = 0$. If the Lebesgue measure was positive, i.e., $\mu(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) > 0$, then the probability of this set would also be non-zero, i.e., $\mathbb{P}(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) > 0$. The main insight to show this follows from the observation that the probability density is positive on the set $\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e$ since the set is part of the support of Z_{inv}^e.

A formal argument to show $\mu(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) > 0 \implies \mathbb{P}(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) > 0$ goes as follows.

Assume the contrary, i.e., $\mathbb{P}(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) = 0$. Let the density be denoted as $f_{Z_{\mathsf{inv}}^e}$. Define the set $\mathcal{P}_k = \{z_{\mathsf{inv}} \in \mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e \mid f_{Z_{\mathsf{inv}}^e}(z) > \frac{1}{k}\}$.

$$\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e = \cup_{k=1}^{\infty} \mathcal{P}_k \tag{2.8.56}$$

$\mathcal{P}_k \uparrow \mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e \implies \mu(\mathcal{P}_k) \to \mu(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e)$. Since $\mu(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) > 0$, $\exists$ some s for which $\mu(\mathcal{P}_s) > 0$.

Define g_s

$$g_s(x) = \begin{cases} \frac{1}{s} \text{ if } x \in \mathcal{P}_k \\ 0 \text{ otherwise} \end{cases} \tag{2.8.57}$$

$$\mathbb{P}(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) = \int_{\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e} f_{Z_{\mathsf{inv}}^e} d\mu \geq \int_{\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e} g_s d\mu \geq \frac{1}{s}\mu(\mathcal{P}_s) > 0 \tag{2.8.58}$$

$\mu(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) > 0 \implies \mathbb{P}(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) > 0 \implies \mathbb{P}(\tilde{\mathcal{Z}}_{\mathsf{inv}}^e) < 1$ which is a contradiction. Therefore, $\mu(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) = 0$.

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 $\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e$. Recall we assume that $\forall e \in \mathcal{E}_{all}$, Z_{inv}^e 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 $e \in \mathcal{E}_{all}$, $\mu(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) = 0 \implies \mathbb{P}(\mathcal{Z}_{\mathsf{inv}}^e \setminus \tilde{\mathcal{Z}}_{\mathsf{inv}}^e) = 0$. 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 [**7**]. 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 $\frac{1}{|\mathcal{E}_{tr}|}E[(Y^e)^2] = \sigma_Y^2$ and the risk achieved by the ideal $\frac{1}{|\mathcal{E}_{tr}|}E[(N^e)^2] = \sigma_N^2$. If $\sigma_N^2 < r^{\mathsf{th}} < \sigma_Y^2$, 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 $v = \mathbb{E}[\epsilon^e Z_{\mathsf{spu}}^e] \neq 0$, which implies $\mathbb{E}[\epsilon^e X^e] \neq 0$. We show why this is the case next.

$$\mathbb{E}[\epsilon^e X^e] = \mathbb{E}[\epsilon^e S(Z_{\mathsf{inv}}^e, Z_{\mathsf{spu}}^e)] = \mathbb{E}[S\epsilon^e(Z_{\mathsf{inv}}^e, Z_{\mathsf{spu}}^e)] = S(0,v) \neq 0; \text{since } S \text{ is invertible} \tag{2.8.59}$$

The rest of the proof follows from Proposition 17 in [**4**]. 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.

Λ

Remark on invertibility of S. The entire proof extends to the case when S is not invertible but Z_{inv}^{e} 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 (2.8.43) 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 (2.8.43) to the multi-class case (Assumption 10) and follow the same line of reasoning to justify why IB-IRM and IB-ERM succeed.

2.8.7. Derivation of the final objective in equation (2.4.2)

In this section, we give a step-by-step description of derivation of the objective in equation (2.4.2). We rewrite the IB-IRM optimization below in equation (2.8.60).

$$
\begin{aligned}
&\min_{\Phi\in\mathbb{R}^k} \frac{1}{|\mathcal{E}_{tr}|}\sum_e h^e\Big(w\cdot\Phi\Big)\\
&\text{s.t. } \frac{1}{|\mathcal{E}_{tr}|}\sum_e R^e\Big(w\cdot\Phi\Big) \le r^{\mathsf{th}},\\
&1 \in \arg\min_{\tilde{w}\in\mathbb{R}} R^e(\tilde{w}\cdot\Phi).
\end{aligned}
\tag{2.8.60}
$$

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 (2.8.60).

$$
\begin{aligned}
&\min_{\Phi\in\mathbb{R}^k} \frac{1}{|\mathcal{E}_{tr}|}\sum_e h^e\Big(\Phi\Big)\\
&\text{s.t. } \frac{1}{|\mathcal{E}_{tr}|}\sum_e R^e\Big(\Phi\Big) \le r^{\mathsf{th}},\\
&1 \in \arg\min_{\tilde{w}\in\mathbb{R}} R^e(\tilde{w}\cdot\Phi).
\end{aligned}
\tag{2.8.61}
$$

It can be shown that the two forms of optimization in equation (2.8.60) and equation (2.8.61) are equivalent. First, we would like to show that the set of feasible classifiers $w\cdot\Phi$ for the first optimization in equation (2.8.61) and Φ in the second optimization in equation (2.8.61) are the same.

Suppose w^*, Φ^* is a feasible solution to the constraints in equation (2.8.60). Construct $\Phi^\dagger = w^*\cdot\Phi^*$. $\Phi^\dagger$ satisfies the constraint $\frac{1}{|\mathcal{E}_{tr}|}\sum_e R^e\Big(\Phi^\dagger\Big) \le r^{\mathsf{th}}$. Suppose for some environment e, $1 \notin \arg\min_{\tilde{w}} R^e(\tilde{w}\cdot\Phi^\dagger) \implies \exists\, w \neq 1$ such that $R^e(w\cdot\Phi^\dagger) < R^e(\Phi^\dagger)$. If this is the case, then $w\times w^*$ improves over w^* and contradicts the optimality of w^* in equation (2.8.60). This establishes that $\Phi^\dagger$ satisfies the constraints in equation (2.8.60). This shows that the set of feasible classifiers for the first optimization in equation (2.8.60) are a subset of the feasible classifiers in the second optimization (2.8.61).

Suppose Φ^* is a feasible solution to the constraints in equation (2.8.61). Take any scalar w and corresponding representation Φ^*/w. The combined classifier $w\cdot(\Phi^*/w)$ satisfies the first constraint. Suppose $w \notin \arg\min_{\tilde{w}\in\mathbb{R}} R^e(\tilde{w}\cdot\frac{\Phi}{w})$, this implies that $\exists\, w^+ \neq w$ such that $R^e(\frac{w^+}{w}\cdot\Phi^*) < R^e(\Phi^*)$. If this was true, then that contradicts the optimality of 1 in equation (2.8.61). This shows that the set of feasible classifiers for the second optimization in equation (2.8.61) are a subset of the feasible classifiers in the first optimization (2.8.60).

From the above discussion, it is clear that the two formulations result in the same set of feasible $w\cdot\Phi$, which are finally fed into the same entropy minimization objective. Thus the two optimizations are equivalent. To get to the penalized objective in equation (2.4.2) from the equation (2.8.61) there are two key steps: i) converting the invariance constraint into the gradient-based penalty, i.e., the IRMv1 penalty from [**7**], 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 Φ 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 (2.4.1).

$$\sum_e \Big(R^e(\Phi) + \lambda\|\nabla_{w,w=1.0}R^e(w\cdot\Phi)\|^2 + \nu h^e(\Phi)\Big). \tag{2.8.62}$$

where $\|\nabla_{w,w=1.0}R^e(w\cdot\Phi)\|^2$ 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 $h(\Phi)$. We assume that all the environments occur with an equal probability.

$$h(\Phi) = -\mathbb{E}_{X\sim\mathbb{P}}[\log(d\mathbb{P}(\Phi(X))] \tag{2.8.63}$$

where $d\mathbb{P}(\Phi(X))$ is the probability density of predictions (unconditional on the environment), $\mathbb{P} = \frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}}\mathbb{P}^e$ 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

$$h^e(\Phi) = -\mathbb{E}_{X^e\sim\mathbb{P}^e}[\log(d\mathbb{P}^e(\Phi(X^e))], \tag{2.8.64}$$

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

$$h(\Phi|E) = -\frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}}\mathbb{E}[\log(d\mathbb{P}^e(\Phi(X^e))]. \tag{2.8.65}$$

Conditioning reduces entropy $h(\Phi) \geq h(\Phi|E)$ and thus we propose an upper bound on the objective in equation (2.8.62) below

$$\sum_e \Big(R^e(\Phi) + \lambda\|\nabla_{w,w=1.0}R^e(w\cdot\Phi)\|^2 + \nu h(\Phi)\Big). \tag{2.8.66}$$

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

$$\sum_{e} \left(R^{e}(\Phi) + \lambda \|\nabla_{w,w=1.0} R^{e}(w \cdot \Phi)\|^{2} + \gamma \mathsf{Var}(\Phi) \right). \tag{2.8.67}$$

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

In this section, we present a detailed analysis of 2D case in equation (2.3.1) leading up to the proof of Theorem 5. For convenience, we will restate the equation (2.3.1). 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.

$$\begin{aligned}
&Y^e \leftarrow \mathsf{sgn}\Big(X^e_{\mathsf{inv}}\Big), \text{ where } X^e_{\mathsf{inv}} \in \{-1,1\} \text{ is } \mathsf{Bernoulli}\Big(\frac{1}{2}\Big),\\
&X^e_{\mathsf{spu}} \leftarrow X^e_{\mathsf{inv}} W^e, \text{ where } W^e \in \{-1,1\} \text{ is } \mathsf{Bernoulli}\Big(1-p^e\Big) \text{ with selection bias } p^e > \frac{1}{2},
\end{aligned} \tag{2.8.68}$$

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. (2.8.68)), 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 (2.8.68), where the input feature values are continuous. For each $e \in \mathcal{E}_{tr}$ we have

$$\begin{aligned}
X^e_{\mathsf{inv}} &\leftarrow C^e + U^e,\\
Y^e &\leftarrow \mathsf{sgn}(X^e_{\mathsf{inv}}),
\end{aligned} \tag{2.8.69}$$

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

$$X^e_{\mathsf{spu}} \leftarrow C^e + M^e,$$

and with probability p^e,

$$X^e_{\mathsf{spu}} \leftarrow -C^e + M^e,$$

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

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 ($\lambda = 0, \gamma = 0$ in equation (2.4.2)) and we get the following

$$\begin{aligned}
&R_{\mathsf{ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}}) =\\
&\frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}}\Big(p^e e^{-(w_{\mathsf{inv}}+w_{\mathsf{spu}})}\mathbb{E}[e^{-w_{\mathsf{inv}}U^e}e^{-w_{\mathsf{spu}}M^e}] + (1-p^e)e^{-(w_{\mathsf{inv}}-w_{\mathsf{spu}})}\mathbb{E}[e^{-w_{\mathsf{inv}}U^e}e^{w_{\mathsf{spu}}M^e}]\Big)\\
&\mathbb{E}[e^{-w_{\mathsf{inv}}U^e}e^{-w_{\mathsf{spu}}M^e}] = \mathbb{E}[e^{-w_{\mathsf{inv}}U^e}]\mathbb{E}[e^{-w_{\mathsf{spu}}M^e}]\\
&\mathbb{E}[e^{-w_{\mathsf{inv}}U^e}] = \Big(\int_{-\delta}^{\delta} e^{-w_{\mathsf{inv}}u}du\Big)\frac{1}{2\delta} = \frac{e^{w_{\mathsf{inv}}\delta} - e^{-w_{\mathsf{inv}}\delta}}{2w_{\mathsf{inv}}\delta} \approx \frac{(1+w_{\mathsf{inv}}\delta)-(1-w_{\mathsf{inv}}\delta)}{2w_{\mathsf{inv}}\delta} = 1
\end{aligned} \tag{2.8.70}$$

If δ 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 $\{-1{,}1\} \times \{-1{,}1\}$.

$$R_{\mathsf{ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}}) \approx pe^{-(w_{\mathsf{inv}}+w_{\mathsf{spu}})} + (1-p)e^{-(w_{\mathsf{inv}}-w_{\mathsf{spu}})} \tag{2.8.71}$$

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

$$R_{\mathsf{IB-ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}}) \approx pe^{-(w_{\mathsf{inv}}+w_{\mathsf{spu}})} + (1-p)e^{-(w_{\mathsf{inv}}-w_{\mathsf{spu}})} + \gamma[w_{\mathsf{inv}}, w_{\mathsf{spu}}]\Sigma[w_{\mathsf{inv}}, w_{\mathsf{spu}}]^{\mathsf{T}} \tag{2.8.72}$$

where $\Sigma = \begin{pmatrix} 1+\delta^2 & 2p-1 \\ 2p-1 & 1+\delta^2 \end{pmatrix}$. Since δ is small, we approximate Σ as $\begin{pmatrix} 1 & 2p-1 \\ 2p-1 & 1 \end{pmatrix}$.

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

Theorem 10. *Suppose each $e \in \mathcal{E}_{tr}$ follows the 2D case from equation (2.3.1). Set $\lambda = 0$, $\gamma > 0$ in equation (2.4.2) to get the IB-ERM objective with ℓ as exponential loss. Continuous-time gradient descent on this IB-ERM objective achieves $|\frac{w_{\mathsf{spu}}(t)}{w_{\mathsf{inv}}(t)}| \leq \epsilon$ in time less than $\frac{W_0(\frac{1}{2\gamma})}{2(1-p)\epsilon}$ ($W_0(\cdot)$ denotes the principal branch of the Lambert W function), while in the same time the ratio for ERM $|\frac{w_{\mathsf{spu}}(t)}{w_{\mathsf{inv}}(t)}| \geq \ln(\frac{1+2p}{3-2p})/\ln\left(1 + \frac{W_0(\frac{1}{2\gamma})}{2(1-p)\epsilon}\right)$, 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 (2.4.2) for the 2D case.

$$R_{\mathsf{ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}}) = pe^{-(w_{\mathsf{inv}}+w_{\mathsf{spu}})} + (1-p)e^{-(w_{\mathsf{inv}}-w_{\mathsf{spu}})}$$

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

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

$$\begin{aligned} \frac{\partial R_{\mathsf{IB-ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}})}{\partial w_{\mathsf{inv}}} &= -pe^{-(w_{\mathsf{inv}}+w_{\mathsf{spu}})} - (1-p)e^{-(w_{\mathsf{inv}}-w_{\mathsf{spu}})} + 2\gamma(w_{\mathsf{inv}} + (2p-1)w_{\mathsf{spu}}) \\ \frac{\partial R_{\mathsf{IB-ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}})}{\partial w_{\mathsf{spu}}} &= -pe^{-(w_{\mathsf{inv}}+w_{\mathsf{spu}})} + (1-p)e^{-(w_{\mathsf{inv}}-w_{\mathsf{spu}})} + 2\gamma((2p-1)w_{\mathsf{inv}} + w_{\mathsf{spu}}) \end{aligned} \tag{2.8.73}$$

$$\begin{aligned}
&\frac{\partial R_{\mathsf{IB-ERM}}(w_{\mathsf{inv}},w_{\mathsf{spu}})}{\partial w_{\mathsf{inv}}} + \frac{\partial R_{\mathsf{IB-ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}})}{\partial w_{\mathsf{spu}}} = -2pe^{-(w_{\mathsf{inv}}+w_{\mathsf{spu}})} + 4\gamma p(w_{\mathsf{inv}} + w_{\mathsf{spu}}) = 0 \\
&\implies \frac{1}{2\gamma}e^{-(w_{\mathsf{inv}}+w_{\mathsf{spu}})} = w_{\mathsf{inv}} + w_{\mathsf{spu}} \\
&\implies w_{\mathsf{inv}} + w_{\mathsf{spu}} = W_0\Big(\frac{1}{2\gamma}\Big)
\end{aligned} \tag{2.8.74}$$

$$\begin{aligned}
&\frac{\partial R_{\mathsf{IB-ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}})}{\partial w_{\mathsf{inv}}} - \frac{\partial R_{\mathsf{IB-ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}})}{\partial w_{\mathsf{spu}}} = -2(1-p)pe^{-(w_{\mathsf{inv}}-w_{\mathsf{spu}})} + 4\gamma(1-p)(w_{\mathsf{inv}} - w_{\mathsf{spu}}) = 0 \\
&\implies \frac{1}{2\gamma}e^{-(w_{\mathsf{inv}}-w_{\mathsf{spu}})} = w_{\mathsf{inv}} - w_{\mathsf{spu}} \\
&\implies w_{\mathsf{inv}} - w_{\mathsf{spu}} = W_0\Big(\frac{1}{2\gamma}\Big)
\end{aligned} \tag{2.8.75}$$

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

$$\frac{\partial x}{\partial t} = -\Big(\frac{\partial R_{\mathsf{IB-ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}})}{\partial w_{\mathsf{inv}}} + \frac{\partial R_{\mathsf{IB-ERM}}(w_{\mathsf{inv}}, w_{\mathsf{spu}})}{\partial w_{\mathsf{spu}}}\Big) = 2p(e^{-x} - 2\gamma x) \tag{2.8.76}$$

$$\frac{\partial y}{\partial t} = 2(1-p)(e^{-y} - 2\gamma y) \tag{2.8.77}$$

Let us call $x^* = W_0\big(\frac{1}{2\gamma}\big)$; x^* is equilibrium point for both $x(t)$ and $y(t)$. Denote $w_{\mathsf{inv}}(t) = \frac{x(t)+y(t)}{2}$ and $w_{\mathsf{spu}}(t) = \frac{x(t)-sy(t)}{2}$. 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 $(e^{-x} - 2\gamma x)$ is Lipschitz continuous in x on $\mathbb{R}$ the solution to the differential equation exists and is unique for any finite interval $t \in [0, T]$ [**58**]. 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 $V(x - x^*) = (x - x^*)^2$

$$\frac{\partial V(x - x^*)}{\partial t} = 2(x - x^*)\frac{\partial x}{\partial t} = 4p(x - x^*)(e^{-x} - 2\gamma x) \tag{2.8.78}$$

Observe that $\frac{\partial V(x-x^*)}{\partial t} < 0$ for all $x \neq x^*$ and $\frac{\partial V(x-x^*)}{\partial t} = 0$ when $x = x^*$. Therefore, from Lyapunov's asymptotic global stability theorem [**30**] we obtain that $x(t)$ would converge to x^*.

Observe that for $x < x^*$, $\frac{\partial x}{\partial t} > 0$ and moreover $2p(e^{-x} - 2\gamma x)$ is a monotonically decreasing function. For all $x < x^* - \epsilon$, we can bound the rate at which x increases is bounded below by $2p(e^{-x^*+\epsilon} - 2\gamma(x^* - \epsilon)) \approx 2p(e^{-x^*}(1+\epsilon) - 2\gamma x^* + 2\gamma\epsilon) = 2p\epsilon(e^{-x^*} + 2\gamma)$. Let us call $\gamma^* = \epsilon(e^{-x^*} + 2\lambda)$. The rate at which x increases is greater than $2p\epsilon\gamma^*$ and the rate at which y increases is greater than $2(1-p)\epsilon\gamma^*$. Thus the time to convergence for x is atmost $\frac{x^*}{2p\epsilon}$. Similarly, the time to convergence for y is atmost $\frac{x^*}{2(1-p)\epsilon}$. 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 $|x(t) - x^*| \leq \epsilon$ and $|y(t) - x^*| \leq \epsilon$, then $|w_{\mathsf{spu}}(t)| = |\frac{x(t)-y(t)}{2}| = |\frac{x(t)-x^*+x^*-y(t)}{2}| \leq \frac{|x(t)-x^*|}{2} + \frac{|y(t)-x^*|}{2} \leq \epsilon$.

If $|x(t)-x^*| \leq \epsilon$ and $|y(t)-x^*| \leq \epsilon$, then $|w_{\mathsf{inv}}(t)-x^*| = |\frac{x(t)+y(t)}{2} - x^*| = |\frac{x(t)-x^*+y(t)-x^*}{2}| \leq \frac{|x(t)-x^*|}{2} + \frac{|y(t)-x^*|}{2} \leq \epsilon$.

As a result, if $|x(t) - x^*| \leq \epsilon$ and $|y(t) - x^*| \leq \epsilon$, then

$$\frac{|w_{\mathsf{spu}}(t)|}{|w_{\mathsf{inv}}(t)|} \leq \frac{\epsilon}{x^* - \epsilon} \approx \frac{\epsilon}{x^*} \tag{2.8.79}$$

Therefore, to get the ratio $\frac{|w_{\mathsf{spu}}(t)|}{|w_{\mathsf{inv}}(t)|} \leq \frac{\epsilon}{x^*}$ the time taken is at most $\frac{x^*}{2(1-p)\epsilon}$.

In comparison in the same amount of time the ratio $|\frac{w_{\mathsf{spu}}(t)}{w_{\mathsf{inv}}(t)}|$ achieved by gradient descent on R_{ERM} is at least $\frac{\ln(\frac{1+2p}{3-2p})}{\ln(1+\frac{x^*}{2(1-p)\epsilon})}$. The expression for lower bound on the ratio $|\frac{w_{\mathsf{spu}}(t)}{w_{\mathsf{inv}(t)}}|$ is derived by substituting the time taken, i.e., $\frac{x^*}{2(1-p)\epsilon}$, in the expression for the lower bound derived in Section B.3 in [**42**]). Λ

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.

2.8.9. 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 2.2a) and Figure 2.2b).

Example extending the 2D case from equation (2.3.1)**.** For all the environments $e \in \mathcal{E}_{tr}$

$$\begin{aligned} Y^e &\leftarrow X^e_{\mathsf{inv}} \oplus N^e \\ X^{1,e}_{\mathsf{spu}} &\leftarrow Y^e \oplus W^e \\ X^{2,e}_{\mathsf{spu}} &\leftarrow X^e_{\mathsf{inv}} \oplus V^e \end{aligned} \tag{2.8.80}$$

where all the variables in the above SEM are binary {0,1} random variables. $N^e \sim \mathsf{Bernoulli}(q)$, $V^e \sim \mathsf{Bernoulli}(a)$; the distribution of noise N^e and V^e are the same across the environments. $W^e \sim \mathsf{Bernoulli}(u^e)$ where u^e is an environment dependent probability. For all the environments $e \in \mathcal{E}_{all}$, 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 $X^{1,e}_{\mathsf{spu}}$ can change arbitrarily. In this example, observe that $\mathbb{E}[Y^e|X^e]$ varies across the training environments. We show the simplification below.

$$\mathbb{E}[Y^e|X^e] = \mathbb{E}\Big[X^e_{\mathsf{inv}} \oplus N^e \Big| (X^e_{\mathsf{inv}}, X^{1,e}_{\mathsf{spu}}, X^{2,e}_{\mathsf{spu}})\Big] \tag{2.8.81}$$

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

$$\begin{aligned} \mathbb{P}(N^e = 1|X^e_{\mathsf{inv}} = 0, X^{1,e}_{\mathsf{spu}} = 0) &= \frac{\mathbb{P}(N^e = 1, X^e_{\mathsf{inv}} = 0, X^{1,e}_{\mathsf{spu}} = 0)}{\mathbb{P}(N^e = 1, X^e_{\mathsf{inv}} = 0, X^{1,e}_{\mathsf{spu}} = 0) + \mathbb{P}(N^e = 0, X^e_{\mathsf{inv}} = 0, X^{1,e}_{\mathsf{spu}} = 0)} \\ &= \frac{\mathbb{P}(N^e = 1, X^e_{\mathsf{inv}} = 0)u^e}{\mathbb{P}(N^e = 1, X^e_{\mathsf{inv}} = 0)u^e + \mathbb{P}(N^e = 0, X^e_{\mathsf{inv}} = 0)(1-u^e)} \end{aligned} \tag{2.8.82}$$

Note that the above equation (2.8.82) describes the probability computed by the Bayes optimal classifier that relies on input feature dimensions are used. Observe that the above probability in equation (2.8.82) can only be equal across two environments if u^e was the same. Therefore, if $|\mathcal{E}_{tr}| \geq 2$ and the probability u^e varies across the environments, then the invariance constraint restrict us from using the identity representation. However, $\mathbb{E}[Y^e|X^e_{\mathsf{inv}}, X^{2,e}_{\mathsf{spu}}]$ is invariant and so is $\mathbb{E}[Y^e|X^e_{\mathsf{inv}}]$. 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 $X^{2,e}_{\mathsf{spu}}$ as follows

$$\begin{aligned}\hat{\mathbb{P}}(Y^e = 1|X^e_{\mathsf{inv}}, X^{2,e}_{\mathsf{spu}}) = (1-q)\mathsf{I}\Big(w_{\mathsf{inv}}X^e_{\mathsf{inv}} + w_{\mathsf{spu}}X^e_{\mathsf{spu}} - \frac{(w_{\mathsf{inv}} + w_{\mathsf{spu}})}{2}\Big) + \\ q\Big(1 - \mathsf{I}\Big(w_{\mathsf{inv}}X^e_{\mathsf{inv}} + w_{\mathsf{spu}}X^e_{\mathsf{spu}} - \frac{(w_{\mathsf{inv}} + w_{\mathsf{spu}})}{2}\Big)\Big).\end{aligned} \tag{2.8.83}$$

If $w_{\mathsf{inv}} > |w_{\mathsf{spu}}|$, then above classifier $\hat{\mathbb{P}}(Y^e = 1|X^e_{\mathsf{inv}}, X^{2,e}_{\mathsf{spu}})$ matches the true probability distribution conditional on the invariant feature $\mathbb{P}(Y^e = 1|X^e_{\mathsf{inv}})$ on all the training environments and it thus forms a valid invariant predictor with representation that focuses on $X^e_{\mathsf{inv}}, X^{2,e}_{\mathsf{spu}}$. Since the classifier relies on $X^{2,e}_{\mathsf{spu}}$, 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 $X^e_{\mathsf{inv}}, X^{2,e}_{\mathsf{spu}}$ 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 $X^{1,e}_{\mathsf{spu}}$ on top of X^e_{inv} as conditioning on $X^{1,e}_{\mathsf{spu}}$ 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 $e \in \mathcal{E}_{all}$ follows*

$$Y^e \leftarrow \mathsf{I}\Big(w^*_{\mathsf{inv}} \cdot X^e_{\mathsf{inv}}\Big) \oplus N^e \tag{2.8.84}$$

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 $\mathcal{E}_{all}$

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

Assumption 12. *Each environment $e \in \mathcal{E}_{tr}$ follows*

$$\begin{aligned} X^{1,e}_{\mathsf{spu}} &\leftarrow Y^e\mathbf{1} \oplus W^e \\ X^{2,e}_{\mathsf{spu}} &\leftarrow X^e_{\mathsf{inv}} \oplus V^e \end{aligned} \tag{2.8.85}$$

where $\mathbf{1} \in \mathbb{R}^{o'}$ is a vector of ones, $W^e \in \mathbb{R}^{o'}$ is a binary 0-1 vector with each component drawn i.i.d. from $\mathsf{Bernoulli}(u^e)$ vector, V^e is also a binary 0-1 vector with each component drawn i.i.d. from $\mathsf{Bernoulli}(a)$ 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., $|\mathcal{E}_{tr}| \geq 2$.

Assumption 13. *$\mathcal{H}_\Phi$ is a set of diagonal matrices, where each element in the matrix is 0 or 1 ($\mathcal{H}_\Phi$ act as matrices that seletct subset of input features). $\mathcal{H}_w$ is set of all probability distributions on $\mathbb{R}^d$. 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.

$$\begin{aligned} &\min_{\Phi\in\mathcal{H}_\Phi} \frac{1}{|\mathcal{E}_{tr}|}\sum_e H^e(\Phi) \\ &\text{s.t. } \frac{1}{|\mathcal{E}_{tr}|}\sum_e R^e\Big(w\circ\Phi\Big) \leq r^{\mathsf{th}} \\ &\quad w \in \arg\min_{\tilde{w}\in\mathcal{H}_w} R^e(\tilde{w}\circ\Phi) \end{aligned} \tag{2.8.86}$$

Theorem 11. *Suppose the data follows Assumption 11, Assumption 12. Suppose $\mathcal{H}_w$ and $\mathcal{H}_\Phi$ 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 $\tilde{X}_1^e$ of $(X^e_{\mathsf{inv}}, X^{2,e}_{\mathsf{spu}})$ and a subset $\tilde{X}_2^e$ of $X^{1,e}_{\mathsf{spu}}$.

$$\begin{aligned} \mathbb{P}(Y^e=1|\tilde{X}_1^e=0,\tilde{X}_2^e=0) &= \frac{\mathbb{P}(Y^e=1,\tilde{X}_1^e=0,\tilde{X}_2^e=0)}{\mathbb{P}(Y^e=1,\tilde{X}_1^e=0,\tilde{X}_2^e=0)+\mathbb{P}(Y^e=0,\tilde{X}_1^e=0,\tilde{X}_2^e=0)} \\ &= \frac{\mathbb{P}(Y^e=1,\tilde{X}_1^e=0)(u^e)^{o'}}{\mathbb{P}(Y^e=1,\tilde{X}_1^e=0)(u^e)^{o'}+\mathbb{P}(Y^e=1,\tilde{X}_1^e=0)(1-u^e)^{o'}} \end{aligned} \tag{2.8.87}$$

Since $\mathbb{P}(Y^e=1|\tilde{X}_1^e=0,\tilde{X}_2^e=0)$ is strictly monotonic in u^e, this probability cannot be same across two environments. Hence, any $\tilde{X}_1^e,\tilde{X}_2^e$ cannot lead to an invariant predictor across the two environments.

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

$$\begin{aligned} \mathbb{P}(Y^e=1|\tilde{X}^e=0) &= \frac{\mathbb{P}(Y^e=1,\tilde{X}^{1,e}=0)}{\mathbb{P}(Y^e=1,\tilde{X}^{1,e}=0)+\mathbb{P}(Y^e=0,\tilde{X}^{1,e}=0)} \\ &= \frac{\mathbb{P}(Y^e=1)(u^e)^{o'}}{\mathbb{P}(Y^e=1)(u^e)^{o'}+\mathbb{P}(Y^e=0)(1-u^e)^{o'}} \end{aligned} \tag{2.8.88}$$

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 $(X^e_{\mathsf{inv}}, X^{2,e}_{\mathsf{spu}})$. Each subset of $(X^e_{\mathsf{inv}}, X^{2,e}_{\mathsf{spu}})$ 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 $X_{\mathsf{spu}}^{2,e}$. However, when we impose entropy minimization objective, then $X_{\mathsf{spu}}^{2,e}$ 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 (2.8.83) but replacing the hyperplane in the indicator function with hyperplane constructed in Lemma 4.

Chapter 3

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. We propose an approach that combines both these principles and demonstrate its effectiveness on linear unit tests [**10**] and on various high-dimensional real datasets.

In the future, we are interested in finding all the downstream evaluations and measurements that matter and finding that which scales best on all those downstream evaluations simultaneously. We believe with high probability that the model size (number of parameters), the dataset size, and the amount of compute used by the largest (and most economically and scientifically valuable) ML training runs are going to increase drastically over the coming years [**17**]. However, no organization currently has direct access to these larger resources of the future; and it has been empirically verified many, many times (e.g., see Figure 2 (right) of [**63**]) that methods which perform best at smaller scales often are no longer the best performing methods at larger scales. In order to stand the test of time, we now think that it is important to view all downstream evaluations (including OOD Generalization) through the lens of how performance on that downstream evaluation changes as the amount of compute used for training, the dataset size, and the number of model parameters keep increasing.

References

[1] Faruk Ahmed, Yoshua Bengio, Harm van Seijen, and Aaron Courville. Systematic generalisation with group invariant predictions. In *International Conference on Learning Representations*, 2021.

[2] Kartik Ahuja, Karthikeyan Shanmugam, and Amit Dhurandhar. Linear regression games: Convergence guarantees to approximate out-of-distribution solutions. In *International Conference on Artificial Intelligence and Statistics*, pages 1270–1278. PMLR, 2021.

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

[4] Kartik Ahuja, Jun Wang, Amit Dhurandhar, Karthikeyan Shanmugam, and Kush R. Varshney. Empirical or invariant risk minimization? a sample complexity perspective. In *International Conference on Learning Representations*, 2021.

[5] Isabela Albuquerque, João Monteiro, Tiago H Falk, and Ioannis Mitliagkas. Adversarial target-invariant representation learning for domain generalization. *arXiv preprint arXiv:1911.00804*, 2019.

[6] Alexander A Alemi, Ian Fischer, Joshua V Dillon, and Kevin Murphy. Deep variational information bottleneck. *arXiv preprint arXiv:1612.00410*, 2016.

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

[8] Devansh Arpit, Caiming Xiong, and Richard Socher. Entropy penalty: Towards generalization beyond the iid assumption. 2019.

[9] Robert B. Ash and Catherine A. Doléans-Dade. *Probability and Measure Theory.* Academic Press, San Diego, California, 2000.

[10] Benjamin Aubin, Agnieszka Słowik, Martin Arjovsky, Leon Bottou, and David Lopez-Paz. Linear unit-tests for invariance discovery. *arXiv preprint arXiv:2102.10867*, 2021.

[11] Elias Bareinboim, Carlos Brito, and Judea Pearl. Local characterizations of causal Bayesian networks. In *Graph Structures for Knowledge Representation and Reasoning*, pages 1–17. Springer, 2012.

[12] Sara Beery, Grant Van Horn, and Pietro Perona. Recognition in terra incognita. In *Proceedings of the European Conference on Computer Vision*, pages 456–473, 2018.

[13] Shai Ben-David, John Blitzer, Koby Crammer, Alex Kulesza, Fernando Pereira, and Jennifer Wortman Vaughan. A theory of learning from different domains. *Machine learning*, 79(1-2):151–175, 2010.

[14] Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. In *Advances in neural information processing systems*, pages 137–144, 2007.

[15] Shai Ben-David and Ruth Urner. On the hardness of domain adaptation and the utility of unlabeled target samples. In *International Conference on Algorithmic Learning Theory*, pages 139–153. Springer, 2012.

[16] Shiyu Chang, Yang Zhang, Mo Yu, and Tommi S Jaakkola. Invariant rationalization. In *International Conference on Machine Learning, 2020*, 2020.

[17] Ajeya Cotra. Forecasting transformative ai with biological anchors, Sep 2020.

[18] Shai Ben David, Tyler Lu, Teresa Luu, and Dávid Pál. 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, 2010.

[19] Alex J DeGrave, Joseph D Janizek, and Su-In Lee. AI for radiographic COVID-19 detection selects shortcuts over signal. *medRxiv*, 2020.

[20] Zhun Deng, Frances Ding, Cynthia Dwork, Rachel Hong, Giovanni Parmigiani, Prasad Patil, and Pragya Sur. Representation via representations: Domain generalization via adversarially learned invariant representations. *arXiv preprint arXiv:2006.11478*, 2020.

[21] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of Machine Learning Research*, 17(1):2096–2030, 2016.

[22] Vikas Garg, Adam Tauman Kalai, Katrina Ligett, and Steven Wu. Learn to expect the unexpected: Probably approximately correct domain generalization. In *International Conference on Artificial Intelligence and Statistics*, pages 3574–3582. PMLR, 2021.

[23] Robert Geirhos, Jörn-Henrik Jacobsen, Claudio Michaelis, Richard Zemel, Wieland Brendel, Matthias Bethge, and Felix A Wichmann. Shortcut learning in deep neural networks. *Nature Machine Intelligence*, 2(11):665–673, 2020.

[24] Daniel Greenfeld and Uri Shalit. Robust learning with the hilbert-schmidt independence criterion. In *International Conference on Machine Learning*, pages 3759–3768. PMLR, 2020.

[25] Ishaan Gulrajani and David Lopez-Paz. In search of lost domain generalization. In *International Conference on Learning Representations*, 2021.

[26] Christina Heinze-Deml, Jonas Peters, and Nicolai Meinshausen. Invariant causal prediction for nonlinear models. *Journal of Causal Inference*, 6(2), 2018.

[27] Ferenc Huszár. https://www.inference.vc/invariant-risk-minimization/. 2019.

[28] Wengong Jin, Regina Barzilay, and Tommi Jaakkola. Enforcing predictive invariance across structured biomedical domains, 2020.

[29] Pritish Kamath, Akilesh Tangella, Danica J Sutherland, and Nathan Srebro. Does invariant risk minimization capture invariance? *arXiv preprint arXiv:2101.01134*, 2021.

[30] Hassan K Khalil. Lyapunov stability. *Control Systems, Robotics and AutomatioN–Volume XII: Nonlinear, Distributed, and Time Delay Systems-I*, page 115, 2009.

[31] Ilyes Khemakhem, Diederik Kingma, Ricardo Monti, and Aapo Hyvarinen. Variational autoencoders and nonlinear ica: A unifying framework. In *International Conference on Artificial Intelligence and Statistics*, pages 2207–2217. PMLR, 2020.

[32] Andreas Kirsch, Clare Lyle, and Yarin Gal. Unpacking information bottlenecks: Unifying information-theoretic objectives in deep learning. *arXiv preprint arXiv:2003.12537*, 2020.

[33] Masanori Koyama and Shoichiro Yamaguchi. Out-of-distribution generalization with maximal invariant predictor. *arXiv preprint arXiv:2008.01883*, 2020.

[34] David Krueger, Ethan Caballero, Joern-Henrik Jacobsen, Amy Zhang, Jonathan Binas, Dinghuai Zhang, Remi Le Priol, and Aaron Courville. Out-of-distribution generalization via risk extrapolation (rex). *arXiv preprint arXiv:2003.00688*, 2020.

[35] David Lewis. *Counterfactuals.* John Wiley & Sons, 2013.

[36] Ya Li, Xinmei Tian, Mingming Gong, Yajing Liu, Tongliang Liu, Kun Zhang, and Dacheng Tao. Deep domain generalization via conditional invariant adversarial networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 624–639, 2018.

[37] Chaochao Lu, Yuhuai Wu, Jośe Miguel Hernández-Lobato, and Bernhard Schölkopf. Nonlinear invariant risk minimization: A causal approach. *arXiv preprint arXiv:2102.12353*, 2021.

[38] Divyat Mahajan, Shruti Tople, and Amit Sharma. Domain generalization using causal matching. *arXiv preprint arXiv:2006.07500*, 2020.

[39] Toshihiko Matsuura and Tatsuya Harada. Domain generalization using a mixture of multiple latent domains. In *AAAI*, pages 11749–11756, 2020.

[40] Krikamol Muandet, David Balduzzi, and Bernhard Schölkopf. Domain generalization via invariant feature representation. In *International Conference on Machine Learning*, pages 10–18, 2013.

[41] Jens Müller, Robert Schmier, Lynton Ardizzone, Carsten Rother, and Ullrich Köthe. Learning robust models using the principle of independent causal mechanisms. *arXiv preprint arXiv:2010.07167*, 2020.

[42] Vaishnavh Nagarajan, Anders Andreassen, and Behnam Neyshabur. Understanding the failure modes of out-of-distribution generalization. In *International Conference on Learning Representations*, 2021.

[43] Artidoro Pagnoni, Stefan Gramatovici, and Samuel Liu. Pac learning guarantees under covariate shift. *arXiv preprint arXiv:1812.06393*, 2018.

[44] Giambattista Parascandolo, Alexander Neitz, ANTONIO ORVIETO, Luigi Gresele, and Bernhard Schölkopf. Learning explanations that are hard to vary. In *International Conference on Learning Representations*, 2021.

[45] Judea Pearl. Causal diagrams for empirical research. *Biometrika*, 82(4):669–688, 1995.

[46] Judea Pearl. *Causality*. Cambridge university press, 2009.

[47] Jonas Peters, Peter Bühlmann, and Nicolai Meinshausen. Causal inference using invariant prediction: identification and confidence intervals. *arXiv preprint arXiv:1501.01332*, 2015.

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

[49] Mohammad Pezeshki, Sékou-Oumar Kaba, Yoshua Bengio, Aaron Courville, Doina Precup, and Guillaume Lajoie. Gradient starvation: A learning proclivity in neural networks. *arXiv preprint arXiv:2011.09468*, 2020.

[50] Vihari Piratla, Praneeth Netrapalli, and Sunita Sarawagi. Efficient domain generalization via common-specific low-rank decomposition. In *International Conference on Machine Learning, 2020*, 2020.

[51] Ievgen Redko, Emilie Morvant, Amaury Habrard, Marc Sebban, and Younès Bennani. *Advances in Domain Adaptation Theory*. Elsevier, 2019.

[52] Alexander Robey, George J Pappas, and Hamed Hassani. Model-based domain generalization. *arXiv preprint arXiv:2102.11436*, 2021.

[53] Mateo Rojas-Carulla, Bernhard Schölkopf, Richard Turner, and Jonas Peters. Invariant models for causal transfer learning. *The Journal of Machine Learning Research*, 19(1):1309–1342, 2018.

[54] Elan Rosenfeld, Pradeep Ravikumar, and Andrej Risteski. An online learning approach to interpolation and extrapolation in domain generalization. *arXiv preprint arXiv:2102.13128*, 2021.

[55] Elan Rosenfeld, Pradeep Kumar Ravikumar, and Andrej Risteski. The risks of invariant risk minimization. In *International Conference on Learning Representations*, 2021.

[56] Shiori Sagawa, Pang Wei Koh, Tatsunori B Hashimoto, and Percy Liang. Distributionally robust neural networks. In *International Conference on Learning Representations*, 2019.

[57] Bernhard Schölkopf, Dominik Janzing, Jonas Peters, Eleni Sgouritsa, Kun Zhang, and Joris Mooij. On causal and anticausal learning. *arXiv preprint arXiv:1206.6471*, 2012.

[58] George F Simmons. *Differential equations with applications and historical notes*. CRC Press, 2016.

[59] Daniel Soudry, Elad Hoffer, Mor Shpigel Nacson, Suriya Gunasekar, and Nathan Srebro. The implicit bias of gradient descent on separable data. *The Journal of Machine Learning Research*, 19(1):2822–2878, 2018.

[60] DJ Strouse and David J Schwab. The deterministic information bottleneck. *Neural computation*, 29(6):1611–1630, 2017.

[61] Damien Teney, Ehsan Abbasnejad, and Anton van den Hengel. Unshuffling data for improved generalization. *arXiv preprint arXiv:2002.11894*, 2020.

[62] Naftali Tishby, Fernando C Pereira, and William Bialek. The information bottleneck method. *arXiv preprint physics/0004057*, 2000.

[63] Ilya Tolstikhin, Neil Houlsby, Alexander Kolesnikov, Lucas Beyer, Xiaohua Zhai, Thomas Unterthiner, Jessica Yung, Andreas Steiner, Daniel Keysers, Jakob Uszkoreit, et al. Mlp-mixer: An all-mlp architecture for vision. *arXiv preprint arXiv:2105.01601*, 2021.

[64] Vladimir Vapnik. Principles of risk minimization for learning theory. In *Advances in neural information processing systems*, pages 831–838, 1992.

[65] Sang Michael Xie, Ananya Kumar, Robbie Jones, Fereshte Khani, Tengyu Ma, and Percy Liang. In-n-out: Pre-training and self-training using auxiliary information for out-of-distribution robustness. In *International Conference on Learning Representations*, 2021.

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

[67] Han Zhao, Remi Tachet des Combes, Kun Zhang, and Geoffrey J Gordon. On learning invariant representation for domain adaptation. *arXiv preprint arXiv:1901.09453*, 2019.

[68] Shanshan Zhao, Mingming Gong, Tongliang Liu, Huan Fu, and Dacheng Tao. Domain generalization via entropy regularization. *Advances in Neural Information Processing Systems*, 33, 2020.