The Missing Invariance Principle found – the Reciprocal Twin of Invariant Risk Minimization

Anonymous Author(s)
Affiliation
Address
email

Abstract

Machine learning models often generalize poorly to out-of-distribution (OOD) data as a result of relying on features that are spuriously correlated with the label during training. Recently, the technique of Invariant Risk Minimization (IRM) was proposed to learn predictors that only use invariant features by conserving the feature-conditioned class expectation $\mathbb{E}_e[y|f(x)]$ across environments. However, more recent studies have demonstrated that IRM can fail in various task settings. Here, we identify a fundamental flaw of IRM formulation that causes the failure. We then introduce a complementary notion of invariance, MRI, that is based on conserving the class-conditioned feature expectation $\mathbb{E}_e[f(x)|y]$ across environments, that corrects for the flaw in IRM. Further, we introduce a simplified, practical version of the MRI formulation called as MRI-v1. We note that this constraint is convex which confers it with an advantage over the practical version of IRM, IRM-v1, which imposes non-convex constraints. We prove that in a general linear problem setting, MRI-v1 can guarantee invariant predictors given sufficient environments. We also empirically demonstrate that MRI strongly out-performs IRM and consistently achieves near-optimal OOD generalization in image-based nonlinear problems.

1 Introduction

Deep learning models have shown tremendous success over the past decade. These models show great generalization properties when tested on the same distribution as the training dataset (in-distribution generalization). However, these models often show catastrophic failure when tested on out-of-distribution dataset, revealing that they learn the shortcut to use easier-to-fit features that are spuriously correlated to the label in the given training domains but don't generalize to the testing domains. For example, deep networks trained on pictures of cow with only grassy backgrounds in the training domain will use the background color as the predictive feature which is easier to learn and generalize poorly to pictures of cow, for example, with a dessert background.

Recently, there has been a growing interest in developing models that generalize well across multiple domains. In particular, there has been a recent body of works that focus on developing algorithms that attempt to learn invariant predictors (Arjovsky et al., 2019; Ahuja et al., 2021; Peters et al., 2016; Rojas-Carulla et al., 2018; Heinze-Deml et al., 2018). Invariant Risk Minimization (IRM) , in particular, has attracted a lot of attention as it is one of the first methods to be compatible with deep learning techniques. However, several follow-up studies have demonstrated the unreliability of IRM at learning invariant representations (Kamath et al., 2021; Rosenfeld et al., 2020; Gulrajani and Lopez-Paz, 2020; Ahuja et al., 2020, 2021). While many of these studies have analyzed various limitations of IRM and IRM-v1 (practical version of IRM), we argue here that a major flaw in the IRM formulation has been overlooked and we propose a method to eliminates this flaw.

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

Related works There has been considerable work in the field of learning invariant representations. They vary from learning domain-invariant feature representations conserving $P(f(x))$ using kernel methods (Muandet et al., 2013; Ghifary et al., 2016; Hu et al., 2020), variational autoencoder (Ilse et al., 2020), and adversarial networks Ganin et al. (2016); Long et al. (2018); Akuzawa et al. (2019); Albuquerque et al. (2020) to learning invariant class-conditional features $P(f(x)|y)$ (Gong et al., 2016; Li et al., 2018b) in the context of domain adaptation, which assumes access to the test distribution for adaptation. There is also a large body of work to learn invariant representations in the field of domain generalization that doesn't assume access to test distributions. This includes imposing invariance of $\mathbb{E}_e[y|f(x)]$ (Arjovsky et al., 2019) with information bottleneck constraint (Ahuja et al. (2021)), imposing object-invariant condition (Mahajan et al. (2021)), using domain inference (Creager et al., 2021), model calibration (Wald et al., 2021), and others (Krueger et al., 2021; Li et al., 2018a; Shankar et al., 2018).

Our Contributions. We reformulate IRM in a simple more generalizable way, and show that it can be modified to yield a new complementary notion of invariance, called MRI. We also show that IRM has a fundamental flaw due to an extraneous constraint which causes it to fail. MRI is shown to be free of this flaw. We prove that in a general linear problem setting, MRI-v1 can guarantee invariant predictors given sufficient environments. We then empirically demonstrate that MRI strongly outperforms IRM and consistently achieves near-optimal OOD generalization in image-based nonlinear problems. [1]

2 Problem Formulation

We consider a set of environments $\mathcal{E} = \{e\}$, each of which defines a distribution $P_e(x, y)$ over inputs and labels, from which the dataset of the environment is drawn $\mathcal{D}_e \equiv \{(x_j^e \in \mathbb{R}^d, y_j^e \in \mathbb{R})\}$. The distribution $P_e(x, y)$ is assumed to be generated according to the causal graph introduced in (Rosenfeld et al., 2020) (Fig.1), which includes latent features that are invariantly z_i or spuriously z_s correlated with the environment. The observation x is generated as a function of the latent features: $x = g(z_i, z_s)$ is assumed to be injective, which makes it possible to recover the latent features from the observation.

In this formulation, we have using d-separation that the label is independent from the environment given the invariant latent feature: $y \perp\!\!\!\perp e|z_i$ and the invariant latent feature is independent from the environment given the label: $z_i \perp\!\!\!\perp e|y$.

The risk of f in environment e is defined as the population average

$$\begin{aligned}
\mathcal{L}_e(f) &\equiv \mathbb{E}_{P_e(x,y)}[l(f(x), y)] && (1)\\
&= \mathbb{E}_{P_e(o)}[\mathbb{E}_{P_e(y|o)}[l(o, y)]] && (2)\\
&= \mathbb{E}_{P(y)}[\mathbb{E}_{P_e(o|y)}[l(o, y)]] && (3)
\end{aligned}$$

where $o = f(x)$ is the predictor's output. Here, we consider standard convex loss functions $l : O \times Y \to \mathbb{R}_{\geq 0}$, including the square loss $l_{\text{sq}}(o, y) = \frac{1}{2}\|o - y\|^2$, and the binary-cross-entropy loss $l_{\text{BCE}}(o, y) = -(1 + y)\log((1 + \tanh(o/2))/2) - (1 - y)\log((1 - \tanh(o/2))/2)$ (for $Y = \{-1, 1\}$)[2].

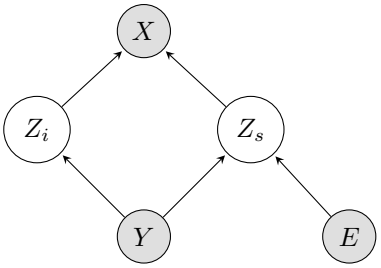

Figure 1: Causal graph

[1] Code available at `https://anonymous.4open.science/r/MRI-2608/readme`.

[2] This is equivalent to the better known form $l_{\text{BCE}}(o, y) = -y\log(\sigma(o)) - (1 - y)\log(1 - \sigma(o))$ for $Y = \{0, 1\}$, where $\sigma(o) = 1/(1 + e^{-o})$.

3 IRM vs MRI invariance

3.1 IRM paradigm

The core idea underlying IRM is the invariance of feature-conditioned label expectation $\mathbb{E}_e[y|o] \equiv \mathbb{E}_{P_e(y|o)}[y]$ across environments, i.e.

$$\forall e_1, e_2 \in \mathcal{E}, \qquad \mathbb{E}_{e_1}[y|o] = \mathbb{E}_{e_2}[y|o]. \tag{4}$$

The IRM paradigm (Arjovsky et al., 2019) proposes to impose this invariance via the following formulation:

Definition 3.1. A representation $f : X \to Z$ is IRM invariant over a set of environments $\mathcal{E}$ if there exists $\psi : Z \to O$ such that ψ is simultaneously optimal on f for all environments $e \in \mathcal{E}$

$$\forall e \in \mathcal{E}, \qquad \psi \in \arg\min_{\bar{\psi}} \mathcal{L}_e(\bar{\psi} \circ f) \tag{5}$$

where ψ is assumed to be unrestricted in the space of all measurable functions.

This definition has shown to be equivalent to the following condition (Kamath et al., 2021),

$$\forall e \in \mathcal{E}, \qquad \mathbb{E}_e[y|o] = \sigma(\psi(o)). \tag{6}$$

where $\sigma(\cdot)$ is a monotonic function such that $\sigma(o) = o$ for square loss, $\sigma(o) = \tanh(o/2)$ for BCE loss. Since the RHS is constant with respect to environments, eq (6) indeed implies eq (4). However, there is a critical difference: Eq (4) only requires the conditional expectation $\mathbb{E}_e[y|o]$ to share the same value across environments, *i.e.*, it imposes only $|\mathcal{E}| - 1$ number of constraints, whereas eq (6) requires that value to be also equal to the RHS term imposing $|\mathcal{E}|$ constraints. This extra constraint originates from the fact that IRM Eq (5) imposes 1 constraint for each environment.

Perturbation-based IRM Definition 3.1 is overly complex due to the presence of ψ and its optimality condition. For further analysis, it helps to simplify the definition by absorbing ψ into the predictor f and considering arbitrary perturbations on the output, which yields the following reformulation of IRM.

Definition 3.2. A predictor $f : X \to O$ is IRM invariant over a set of environments $\mathcal{E}$, if the risk $\mathcal{L}_e(f)$ is preserved under arbitrary infinitesimal perturbations on its output $o = f(x)$ for all environments, *i.e.*

$$\begin{aligned} \forall e \in \mathcal{E}, \forall\, \delta\psi, \qquad \delta\mathcal{L}_e(f) &\equiv \lim_{\epsilon \to 0} \mathbb{E}_{P_e(o)}[\mathbb{E}_{P_e(y|o)}[l(o + \epsilon\delta\psi(o), y) - l(o, y)]]/\epsilon \\ &= \mathbb{E}_{P_e(o)}[\mathbb{E}_{P_e(y|o)}[\partial_o l(o, y)] \cdot \delta\psi(o)] = 0 \end{aligned} \tag{7}$$

where $\delta\psi(\cdot)$ is an arbitrary perturbation that is unrestricted in the space of all measurable functions.

Lemma 3.3. *This reformulated definition 3.2 is equivalent to eq (6).*

Proof. Eq (7) is satisfied if and only if $\mathbb{E}_{P_e(y|o)}[\partial_o l(o, y)] = 0$. Note that $\partial_o l_{\text{sq}}(o, y) = o - y$ and $\partial_o l_{\text{BCE}}(o, y) = \tanh(o/2) - y$. Therefore, we arrive at

$$\forall e \in \mathcal{E}, \qquad \mathbb{E}_e[y|o] = \sigma(o). \tag{8}$$

□

IRM-v1 Due to the impracticality of considering optimal classifiers in the space of unrestricted functions, Arjovsky et al. (2019) suggested restricting the functional class of ψ to the space of linear functions. In definition 3.2, this corresponds to restricting the output perturbations to the space of linear functions, or equivalently, an identity function: *i.e.* $\delta\psi(o) = o$, which reduces eq (7) to

$$\begin{aligned} \delta\mathcal{L}_e(f) &= \mathbb{E}_{P_e(o)}[\mathbb{E}_{P_e(y|o)}[\,\partial_o l(o, y) \cdot o\,] \\ &= \mathbb{E}_e[o\sigma(o)] - \mathbb{E}_e[oy] = 0 \end{aligned} \tag{9}$$

This reduced formulation is called IRM-v1 in Arjovsky et al. (2019) or IRM-s in Kamath et al. (2021).

Note that the first term $\mathbb{E}_e[o\sigma(o)]$ of eq (9) originates from the RHS of eq (6). If this term were constant, it would yield the conservation of $\mathbb{E}_e[oy] = \mathbb{E}_{P_e(o)}[\mathbb{E}_e[y|o] \cdot o]$, a necessary condition for the conservation of $\mathbb{E}_e[y|o]$ in eq (4). However, since the constancy of this term is generally not guaranteed, eq (9) does not necessarily prescribe any meaningful invariance. Thus, IRM-v1 is not consistent with IRM's invariance. Although this inconsistency was empirically shown in Kamath et al. (2021), the actual cause of the problem was never discussed. Our analysis reveals that the root cause originates from the extraneous constraint of IRM.

	Square loss ($\delta\mathcal{L}_e(f)$)	BCE loss ($\delta\mathcal{L}_e(f)$)	Constraint
IRM	$\mathbb{E}_e[y\|o] - o$	$\mathbb{E}_e[y\|o] - \sigma(o)$	$\delta\mathcal{L}_e(f) = 0$
MRI	$\mathbb{E}_e[o\|y] - y$	$\mathbb{E}_e[o\|y]$	$\delta\mathcal{L}_e(f)$ is conserved
IRM-v1	$\mathbb{E}_e[oo - oy]$	$\mathbb{E}_e[o\sigma(o) - oy]$	$\delta\mathcal{L}_e(f) = 0$
MRI-v1	$\mathbb{E}_e[oy - yy]$	$\mathbb{E}_e[oy]$	$\delta\mathcal{L}_e(f)$ is conserved

Table 1: Summary of invariance constraints. $\sigma(o) = \tanh(o/2)$.

3.2 MRI paradigm

We now introduce a complementary notion of invariance by slightly modifying definition 3.2, which we call the *Mirror Reflected IRM*, or MRI.

Definition 3.4. A predictor $f : X \to O$ is MRI invariant over a set of environments $\mathcal{E}$, if for all environments $\mathcal{L}_e(f)$ gets *identically affected* by arbitrary infinitesimal perturbations on label, *i.e.*

$$\forall e_1, e_2 \in \mathcal{E}, \qquad \delta\mathcal{L}_{e_1}(f) = \delta\mathcal{L}_{e_2}(f) \tag{10}$$

$$\text{where} \qquad \delta\mathcal{L}_e(f) \equiv \lim_{\epsilon\to 0} \mathbb{E}_{P(y)}[\mathbb{E}_{P_e(o|y)}[\, l(o, y + \epsilon\delta\psi(y)) - l(o, y)\,]]/\epsilon$$

$$= \mathbb{E}_{P(y)}[\mathbb{E}_{P_e(o|y)}[\,\partial_y l(o, y)] \cdot \delta\psi(y)\,] \tag{11}$$

where $\delta\psi(\cdot)$ is an arbitrary perturbation that is unrestricted in the space of all measurable functions.

Lemma 3.5. *Definition 3.4 implies the conservation of* $\mathbb{E}_e[o|y] \equiv \mathbb{E}_{P_e(o|y)}[o]$ *across environments,*

$$\forall e_1, e_2 \in \mathcal{E}, \qquad \mathbb{E}_{e_1}[o|y] = \mathbb{E}_{e_2}[o|y]. \tag{12}$$

Proof. Eq (10) is satisfied if and only if $\mathbb{E}_e[\partial_y l(o, y)|y]$ is conserved. Since $\partial_y l_{\text{sq}}(o, y) = y - o$ and $\partial_y l_{\text{BCE}}(o, y) = -o$, this yields eq (12). □

Note that unlike IRM, MRI prescribes $E - 1$ constraints, which is the right number of constraints.

MRI-v1 Restricting the label perturbations to the space of linear functions, or equivalently, an identity function $\delta\psi(y) = y$, reduces the MRI formulation eq (11) to

$$\delta\mathcal{L}_e(f) = \mathbb{E}_{P_e(o,y)}[\,\partial_y l(o, y) \cdot y\,], \tag{13}$$

which yields the following conservation law

$$\forall e_1, e_2 \in \mathcal{E}, \qquad \mathbb{E}_{e_1}[oy] = \mathbb{E}_{e_2}[oy], \tag{14}$$

which is indeed a necessary condition for MRI invariance eq (12). Moreover, MRI-v1 eq (14) imposes linear equality constraint with respect to the output o. The consistency with MRI invariance, as well as the convexity of constraint, provides MRI-v1 with great practical advantages over IRM-v1, which we discuss in following sections.

3.3 Objective

The full methods of IRM-v1 and MRI-v1 involve solving the following constrained optimization

$$\min_{\theta} \bar{\mathcal{L}}_{\text{tr}}(f(\theta)) \qquad \text{subject to} \quad \vec{c}(f(\theta)) = \vec{0}, \tag{15}$$

where the objective $\bar{\mathcal{L}}_{\text{tr}}(f) = \frac{1}{|\mathcal{E}_{tr}|}\sum_{e\in\mathcal{E}_{tr}} \mathcal{L}_e(f)$ is the average risk over the set of training environments $\mathcal{E}_{tr} \subset \mathcal{E}$, θ is the parameters of the predictor f, and the vector-valued constraint function contains the perturbed risks $\delta\mathcal{L}_e(f), \forall e \in \mathcal{E}_{tr}$ for the case of IRM-v1 as the perturbed risk itself, or the difference between pairs of perturbed risks for the case of MRI-v1 (See Table 1).

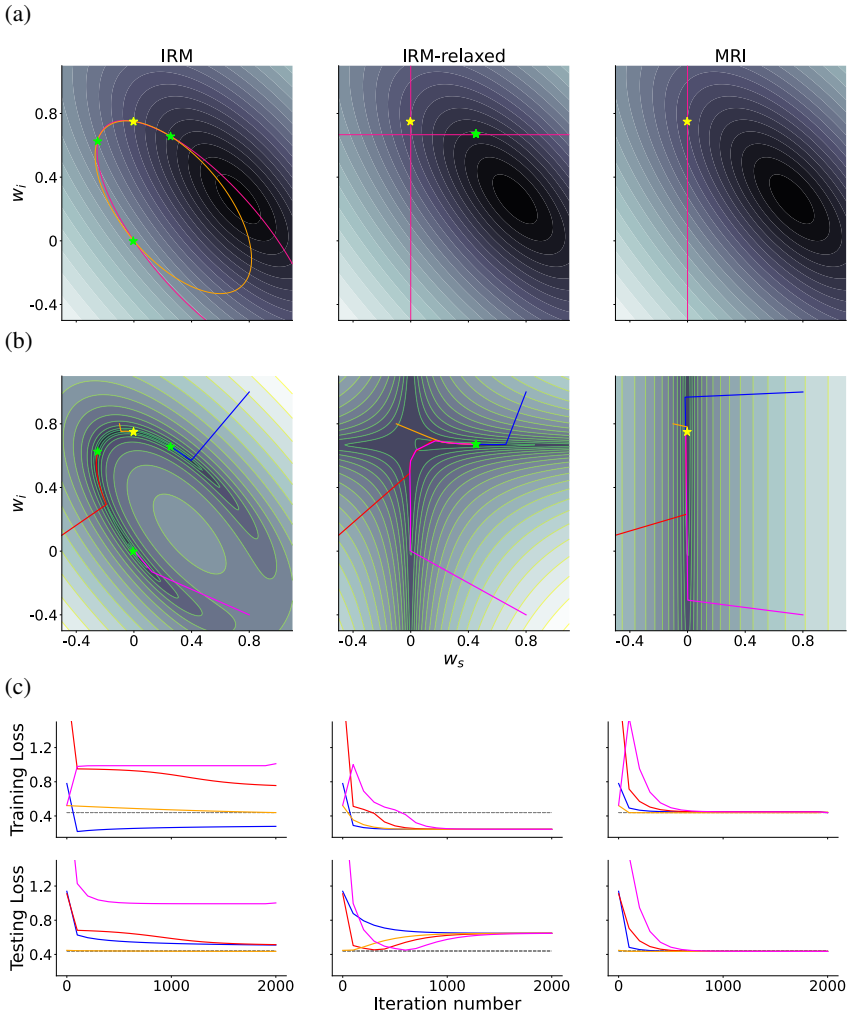

Figure 2: Shape-texture linear regression problem (Section 5.2) for a linear predictor model $f(x) = w_i z_i + w_s z_s$. (a) Constraints overlayed with ERM loss, and (b) Soft constraints (penalty method with $\lambda = 5 \times 10^4$) added to ERM loss, (c) Top panel - train loss, bottom panel - test loss.

4 Linear setting

4.1 General linear SEM

In this section, we show that MRI-v1 can correctly remove all spurious features when enough environments are given.

We consider a data generating process according to 1, in which $x = g(z_i, z_s)$ is a linear function. Note that this SEM is a generalization of the linear Gaussian model considered in Rosenfeld et al. (2020), which additionally assumed binary y and a specific noise model (linear additive Gaussian) for the latent features.

We consider a linear predictor $f : X \to O$. Since g is injective and has an inverse over its range, without loss of generality, we can define f as a linear function directly over the latents as

$$o = f(x; \theta) = w_i \cdot z_i + w_s \cdot z_s \tag{16}$$

with $\theta = \{w_i \in \mathbb{R}^{d_i}, w_s \in \mathbb{R}^{d_s}\}$.

Theorem 4.1. *Given $|\mathcal{E}_{tr}| > d_s$ training environments, where d_s is the dimension of the spurious features, and that $\mathbb{E}_e[z_s y]$ are in general linear position, MRI-v1 will eliminate all spurious feature dimensions.*

Proof. MRI-v1 condition:

$$\mathbb{E}_e[oy] = w_i \cdot \mathbb{E}_e[z_i y] + w_s \cdot \mathbb{E}_e[z_s y] = const \quad \forall e \in \mathcal{E} \tag{17}$$

This can be written in a matrix form as

$$w_s \cdot M = c' \cdot \vec{1}, \tag{18}$$

where $M \equiv [\mathbb{E}_e[z_s y]]_{e \in \mathcal{E}_{tr}} \in \mathbb{R}^{d_s \times |\mathcal{E}_{tr}|}$, and $c' \equiv const - w_i \cdot \mathbb{E}_e[z_i y]$. Since $rank(M) = d_s$, Eq (18) is true if and only if $w_s = 0$ and $c' = 0$.

□

Rosenfeld et al. (2020) derives a similar result to that of our Theorem 4.1 stating that IRM-v1 constraint can eliminate all spurious feature dimensions given $|\mathcal{E}_{tr}| > d_s$. However, unlike our results for MRI, their result is restricted to a specialized linear family of environments with binary labels and additive gaussian noise. In fact, IRM-v1 is known to fail to eliminate spurious feature dimensions in a more general linear problems in which the noise is not additive Gaussian (Arjovsky et al., 2019; Kamath et al., 2021).

4.2 Shape-texture Linear Regression

Here, we consider a linear regression problem on the Shape-texture dataset which has 2 training environments and include 1 invariant and 1 spurious features. The full analytical solutions are presented in the Supplementary Materials.

IRM-v1 IRM-v1 has two quadratic equality constraints, shown as two ellipses in Fig 2a. The intersection between 2 non-convex constraints on a 2 dimensional problem leaves the solution space to be a disjoint set of (0-d) points, shown as 4 stars. Note that among the 4 solutions, only two of them correspond to an invariant ($w_s = 0$) predictor, one of which is the trivial zero predictor, while the other two are non-invariant, which confirms the inconsistency of IRM-v1 at identifying invariant predictors (Kamath et al., 2021; Arjovsky et al., 2019).

MRI-v1 In contrast, consistent with Theorem 4.1, MRI-v1 has a single linear equality constraint (Fig 2a), which exactly yields the set of invarian solutions $w_s = 0$. That is, every point on the constraint line is a invariant predictor. Since this is a convex constraint, the convex objective $\bar{\mathcal{L}}_{\text{tr}}(f)$ has a unique constrained optimum, which yields the invariant optimum predictor (Fig 2a).

IRM-relaxed To isolate the non-convexity issue of the IRM-v1 constraint, we consider a relaxed version of IRM-v1 such that $\delta\mathcal{L}_{e_1}(f) = \delta\mathcal{L}_{e_2}(f)$. Since it has only a single constraint, it has 1d continuous set of points as possible solutions (Fig 2a). However, it is still a nonconvex constraint. In addition to the $w_s = 0$ line, it also exhibits $w_i = 2/3$ line as the solution (Fig 2a). As a result, the $\bar{\mathcal{L}}_{\text{tr}}(f)$ objective exhibits two constrained optimum solutions, only one of which is invariant (Fig 2a).

Learning Dynamics In Fig 2(b,c), we show simulation of gradient descent learning dynamics of the objective eq (19) ($\mu = 5 \times 10^4$). The convergence pattern for IRM-v1 is highly dependent on the initialization, due to the presence of multiple local minima. In this case, the invariant optimum is especially extremely difficult to converge to as it has higher risk than the nearby non-invariant minima. In contrast, MRI-v1 always converges to the invariant optimum regardless of initialization since it is a unique minimum. For IRM-relaxed, the invariant optimum point is lost at this value of μ, which exists in the limit $\mu \to \infty$. As a result, all trajectories converge to the non-invariant optimum, which has lower training risk. We show similar results for other linear problems including shape-texture classification and toy-CMNIST in Supplementary Materials.

5 Non-linear Setting

5.1 Methods: Constrained Optimization

Solving constrained optimization numerically requires converting the problem into an unconstrained optimization problem such that the constraint is imposed as a soft regularization term. This allows the use of any off-the-shelf gradient-based optimization algorithm.

Penalty Method (PM) Penalty method is the most commonly used approach, including in Arjovsky et al. (2019), which adds the squared constraint as a penalty term to the objective,

$$\min_{\theta} \lim_{\mu \to \infty} \bar{\mathcal{L}}(f(\theta)) + \mu \left\| \vec{c}(f(\theta)) \right\|^2 \tag{19}$$

However, this method requires large values for μ to approximate the exact constraint, which leads to training instability and slow convergence Bertsekas (1976).

Augmented Lagrangian Method (ALM) ALM was introduced to overcome the limitations of penalty method (Bertsekas, 1976), which adds a Lagrange multiplier term $\vec{\lambda}$ to (19)

$$\min_{\theta} \bar{\mathcal{L}}(f(\theta)) + \mu \left\| \vec{c}(f(\theta)) \right\|^2 + \vec{\lambda} \cdot \vec{c}(\theta). \tag{20}$$

This method involves updating λ^i at each training iteration t to accumulate the residual constraint, *i.e.* $\vec{\lambda}_e^{t+1} = \lambda_e^t + \mu \vec{c}(\theta^t)$, and then updating θ^t according to a gradient descent step with respect to the combined objective in eq (20). ALM quickly converges to exactly satisfy the constraints with moderate values of μ (~ 10) without fine tuning, and thus exhibits much faster and stable convergence Bertsekas (1976). $\vec{\lambda}$ is typically initialized at $\vec{0}$, which has an annealing effect on ALM. Moreover, ALM has an interesting effect when used with mini-batches, in which the constraints have slight mismatches across iterations. Since the Lagrange multiplier λ is accumulated over iterations, it imposes the model to satisfy an averaged constraint. In contrast, PM penalizes the model to satisfy all of the conflicting constraints, and thus decaying the model toward the zero-predictor. Unless noted otherwise, we used ALM by default for our optimization.

5.2 Datasets

5.2.1 Shape-Texture Dataset

We develop a new versatile dataset designed for domain generalization, called the Shape-Texture dataset, based on the data generating model in Fig 1. This dataset is designed to test algorithms on linear and non-linear regression and classification problems in multiple adjustable settings.

The dataset involves an invariant latent feature $\theta_i \in \mathbb{R}$, a spurious latent feature $\theta_s \in \mathbb{R}$, and the latent feature $\theta_y \in \mathbb{R}$ for the label. Each of them represents an angle of orientation, and therefore is naturally represented on a unit circle in the complex plane as $e^{i\theta} \in \mathcal{S}^1$. In the linear regression setting, the observed input is simply the concatenated features $x = [e^{i\theta_i}, e^{i\theta_s}]$, and the label is $y = e^{i\theta_y}$. Whereas in the linear classification setting, the input is given by $x = [\sin(\theta_i), \sin(\theta_s)]$, and the label is $y = H(\sin(\theta_y))$ where H denotes the Heavyside step function. In the non-linear regression and classification setting, the label remains the same as their respective linear versions but the input is an image (Fig 7) composed of two planar waves of differing frequencies; The orientation of the low frequency wave (a.k.a shape) is given by $\theta_i/2$, and the orientation of the high frequency wave (a.k.a texture) is given by $\theta_s/2$. For all task settings, the latent angles and the label angle are linearly related as $\theta_y \leftarrow \theta_i + \xi_o, \theta_s \leftarrow \theta_y + \xi_e$, where $\xi_* \sim \mathcal{N}(0, \sigma_*)$ with probability p_*, or $\xi_* \sim \mathcal{U}(-\pi, \pi)$ probability $1 - p_*$ for $* \in \{i, s\}$. p_i, σ_i are fixed invariant parameters whereas p_s, σ_s can vary from one environment to another. In this work, for all task settings, we use two training environments and one testing environment with $p_i = 0.75$ for all environments, $p_s = 0.8, 1$ for the two training environments, $p_s = 0$ for the testing environment, and $\sigma_i = \sigma_s = 0$ for all environments.

Result We use a small convolutional network for the image-based regression and classification dataset. In addition to reporting the results for IRM-v1 and MRI-v1 algorithms, we report the performance of the ERM algorithm, which minimizes the average risk across environments (Table 2).

We also report the performance of the Oracle (Table 2), where the Oracle is trained on environments in which the spurious features are not correlated with the label. For both regression and classification setting, we find that IRM-v1 it is better than ERM although there is a significant generalization error (Table 2). We find similar results for accuracy (see Table 3). However, MRI-v1 gives almost identical performance across training and testing environments irrespective of the network initialization (performs as well as Oracle). As a result, MRI-v1 achieve a lower testing loss than IRM-v1 and ERM in both task settings (Table 2). We find similar results for accuracy (see Table 3).

5.2.2 Colored MNIST

The Colored MNIST (CMNIST) dataset was introduced by Arjovsky et al. (2019) to analyze IRM-v1's ability to learn non-linear invariant predictors. This is a synthetic binary classification task derived from MNIST. In this dataset, the color of the image is determined in correlation with the label whose correlation probability p_s changes between environments. The correlation probability p_i between digit and label remains the same across all environments. Therefore, in this dataset color is the spurious feature and digit is the invariant feature. This dataset follows the data generating model in Fig 1.

In this work, we simulate two versions of the Colored MNIST dataset corresponding to two different sets of correlation probabilities p_s and p_i. The first version, CMNISTa, uses the same correlation probabilities as Arjovsky et al. (2019) with $p_i = 0.75$ for all environments, $p_s = 0.8, 0.9$ for the two training environments, and $p_s = 0.1$ for the testing environment. For the other version, CMNISTb, we use $p_i = 0.95$ for all environments, $p_s = 0.8, 1$ for the two training environments, and $p_s = 0.1$ for the testing environment. We also simulate the linear toy versions, toy-CMNIST (also known as the two-bit environment Kamath et al. (2021)), of both these datasets in which the image input is replaced by $x = [X_1, X_2]$, where $X_1 \in \{-1, 1\}$ corresponds to the label of the digit and $X_2 \in \{-1, 1\}$ corresponds to the color.

Result For both the CMNIST datasets, we use the LeNet architecture (LeCun et al., 1998). We find that both MRI-v1 and IRM-v1 perform better than ERM for both datasets. MRI-v1 again performs identically across all environments and almost as well the Oracle for both datasets (Table 2). IRM-v1, however, has slightly worse OOD generalization performance than MRI-v1 for CMNISTa, but it is much worse for the CMNISTb dataset (Table 2). This result is consistent with the results for the linear toy-CMNIST datasets. We find that the IRM-v1 constraint does not contain a non-invariant solution for toy-CMNISTa, whereas for toy-CMNISTb it does contain non-invariant solutions (Fig 6). However, MRI-v1 constraints the predictor to ignore the spurious feature ($w_s = 0$) in the toy version of both the datasets (Fig 6) and therefore, it should converge to the invariant solution in both cases. Results for accuracy are consistent with the results of the linear toy version as well (see Table 3)).

In conclusion, we confirm that the results from the linear settings also apply to practical training of deep networks in non-linear image-based regression and classification tasks. MRI-v1 provides best OOD generalization across all tasks. In fact, it performs almost as well as the Oracle.

	Oracle		MRI-v1		IRM-v1		ERM	
	Train loss	Test loss	Train loss	Test loss	Train loss	Test loss	Train loss	Test loss
S-T Reg.	0.46 ± 0.0	0.46 ± 0.0	0.45 ± 0.01	$\mathbf{0.46} \pm 0.0$	0.23 ± 0.0	0.62 ± 0.01	0.16 ± 0.0	1.37 ± 0.01
S-T Cls.	0.47 ± 0.0	0.47 ± 0.0	0.47 ± 0.0	$\mathbf{0.47} \pm 0.0$	0.4 ± 0.01	0.51 ± 0.01	0.24 ± 0.0	1.16 ± 0.01
CMNISTa	0.57 ± 0.0	0.58 ± 0.0	0.63 ± 0.02	$\mathbf{0.64} \pm 0.01$	0.62 ± 0.02	0.69 ± 0.02	0.36 ± 0.0	1.44 ± 0.01
CMNISTb	0.22 ± 0.0	0.24 ± 0.01	0.25 ± 0.01	$\mathbf{0.29} \pm 0.01$	0.17 ± 0.01	0.47 ± 0.02	0.13 ± 0.0	0.73 ± 0.01

Table 2: Loss comparison for image-based experiments. Oracle uses training environments in which the spurious features are not correlated with the label.

6 Discussion

In this work, we introduce the MRI formulation that learns invariant representations using a new notion of invariance based on conserving the class-conditional feature expectation $\mathbb{E}_e[f(x)|y]$ across

environments. We derive a simplified version of the MRI formulation, MRI-v1, that allows us to apply this new notion of invariance to practical training of deep networks. We provide analytical guarantees that MRI will learn invariant predictors under certain conditions. We also demonstrate numerically in multiple task settings, including practical image-based tasks, that MRI provides much better OOD generalization, than IRM (which conserves $\mathbb{E}_e[y|f(x)]$) and ERM. In fact, it performs nearly as well as the Oracle.

Disadvantage of IRM formulation. Both the MRI and the IRM formulation are based on seemingly intuitive conservation laws to attain invariance. Why is it then that the MRI formulation is so much more successful at preserving invariance in practice? In this work, we note that the failure of IRM is a side-effect of the extra constraint that requires $\mathbb{E}_e[y|f(x)] = f(x)$, in addition to conserving $\mathbb{E}_e[y|f(x)]$ across environments. Without the extra constraint, the practical linear versions of both the IRM and the MRI formulations match. To our knowledge, this is the first study to identify this major flaw in the IRM formulation.

Convexity matters. Deep learning in general is a highly non-convex optimization problem, yet it doesn't seem to have problem with convergence. It is understood that all minima is just as good as (equivalent to) the global minimum in the highly over-parameterized regime and central to this is the idea of mode-connectivity that suggests that all minima are connected to each other (Draxler et al., 2018; Garipov et al., 2018). In this paper, we suggest that this property is highly dependent on the fact that most loss functions used in deep learning are convex with respect to the network's output. We argue that when loss functions have multiple disconnected local minima in the output space, the overall learning problem with respect to the network parameters can also exhibit disconnected local minima, i.e., broken mode connectivity. To support our argument, we demonstrate that the IRM constraint, which is non-convex in output, leads to multiple disconnected minima. However, the MRI method, which uses a convex constraint, has no problem learning the invariant optima even in highly over-parametrized networks. To our knowledge, this is the first study to discuss the limitations of using constraints that are non-convex in the output.

Finite samples. Although we have defined the algorithms (MRI-v1 and IRM-v1) in terms of population losses, in practice these algorithms are trained with finite sample sizes. Kamath et al. (2021) pointed out that IRM-v1 learns the trivial 0 predictor for high penalty with finite sample sizes. Here, we discuss two additional disadvantages of training with finite sample sizes. Firstly, we find that both MRI-v1 and IRM-v1 are prone to the property of deep networks of memorizing the dataset (Goodfellow et al., 2016) instead of learning patterns. We demonstrate this using a one-hidden-layer ReLU network that learns the linear shape-texture regression dataset. Both the algorithms first learns the correct solution (i.e, invariant solution for MRI-v1 and non-invariant solution for IRM-v1) but then starts memorizing the dataset. However, since this overfitting issue arises as result of finite sample size we can reduce it by increasing the sample size (Fig 4). Additionally, common regularization techniques like mini-batch training (Smith et al. (2021); Fig 4) and weight decay (Goodfellow et al. (2016); Fig 4) can also reduce overfitting. Secondly, we find that due to finite sample size noise in the joint input-label distribution, MRI-v1 learns 'imprecise' invariant predictor (i.e., the training and testing loss don't exactly match) and this precision improves with increase in sample size (Fig 4). To our knowledge, this is the first study to discuss these two disadvantages of learning invariant representation with finite sample sizes.

Limitations. A major limitation of our proposed method is the assumption of no label shift across domains. However, label shift can be easily corrected for. Mahajan et al. (2021) show that class-conditional invariance can fail if $P(z_c|y)$ does not remain the same across domains, which also applies to our method. Like many other invariant representation learning algorithms, including IRM (Ahuja et al., 2020, 2021), our method will be unable to handle covariate shift across domains. Also, like IRM (Ahuja et al., 2021), our method requires that there is significant overlap in the support of different environment distributions.

References

Ahuja, K., Caballero, E., Zhang, D., Bengio, Y., Mitliagkas, I., and Rish, I. (2021). Invariance principle meets information bottleneck for out-of-distribution generalization. *arXiv preprint arXiv:2106.06607*.

Ahuja, K., Wang, J., Dhurandhar, A., Shanmugam, K., and Varshney, K. R. (2020). Empirical or invariant risk minimization? a sample complexity perspective. *arXiv preprint arXiv:2010.16412*.

Akuzawa, K., Iwasawa, Y., and Matsuo, Y. (2019). Adversarial invariant feature learning with accuracy constraint for domain generalization. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pages 315–331. Springer.

Albuquerque, I., Monteiro, J., Darvishi, M., Falk, T. H., and Mitliagkas, I. (2020). Adversarial target-invariant representation learning for domain generalization.

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

Bertsekas, D. P. (1976). Multiplier methods: A survey. *Automatica*, 12(2):133–145.

Creager, E., Jacobsen, J.-H., and Zemel, R. (2021). Environment inference for invariant learning. In *International Conference on Machine Learning*, pages 2189–2200. PMLR.

Draxler, F., Veschgini, K., Salmhofer, M., and Hamprecht, F. (2018). Essentially no barriers in neural network energy landscape. In *International conference on machine learning*, pages 1309–1318. PMLR.

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

Garipov, T., Izmailov, P., Podoprikhin, D., Vetrov, D. P., and Wilson, A. G. (2018). Loss surfaces, mode connectivity, and fast ensembling of dnns. *Advances in neural information processing systems*, 31.

Ghifary, M., Balduzzi, D., Kleijn, W. B., and Zhang, M. (2016). Scatter component analysis: A unified framework for domain adaptation and domain generalization. *IEEE transactions on pattern analysis and machine intelligence*, 39(7):1414–1430.

Gong, M., Zhang, K., Liu, T., Tao, D., Glymour, C., and Schölkopf, B. (2016). Domain adaptation with conditional transferable components. In *International conference on machine learning*, pages 2839–2848. PMLR.

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. MIT Press. http://www.deeplearningbook.org.

Gulrajani, I. and Lopez-Paz, D. (2020). In search of lost domain generalization. *arXiv preprint arXiv:2007.01434*.

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

Hu, S., Zhang, K., Chen, Z., and Chan, L. (2020). Domain generalization via multidomain discriminant analysis. In *Uncertainty in Artificial Intelligence*, pages 292–302. PMLR.

Ilse, M., Tomczak, J. M., Louizos, C., and Welling, M. (2020). Diva: Domain invariant variational autoencoders. In *Medical Imaging with Deep Learning*, pages 322–348. PMLR.

Kamath, P., Tangella, A., Sutherland, D., and Srebro, N. (2021). Does invariant risk minimization capture invariance? In *International Conference on Artificial Intelligence and Statistics*, pages 4069–4077. PMLR.

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

LeCun, Y., Bottou, L., Bengio, Y., and Haffner, P. (1998). Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324.

Li, D., Yang, Y., Song, Y.-Z., and Hospedales, T. M. (2018a). Learning to generalize: Meta-learning for domain generalization. In *Thirty-Second AAAI Conference on Artificial Intelligence*.

Li, Y., Gong, M., Tian, X., Liu, T., and Tao, D. (2018b). Domain generalization via conditional invariant representations. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32.

Long, M., Cao, Z., Wang, J., and Jordan, M. I. (2018). Conditional adversarial domain adaptation. *Advances in neural information processing systems*, 31.

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

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

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

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

Rosenfeld, E., Ravikumar, P., and Risteski, A. (2020). The risks of invariant risk minimization. *arXiv preprint arXiv:2010.05761*.

Shankar, S., Piratla, V., Chakrabarti, S., Chaudhuri, S., Jyothi, P., and Sarawagi, S. (2018). Generalizing across domains via cross-gradient training. *arXiv preprint arXiv:1804.10745*.

Smith, S. L., Dherin, B., Barrett, D. G., and De, S. (2021). On the origin of implicit regularization in stochastic gradient descent. *arXiv preprint arXiv:2101.12176*.

Wald, Y., Feder, A., Greenfeld, D., and Shalit, U. (2021). On calibration and out-of-domain generalization. *Advances in Neural Information Processing Systems*, 34.

Checklist

The checklist follows the references. Please read the checklist guidelines carefully for information on how to answer these questions. For each question, change the default **[TODO]** to [Yes] , [No] , or [N/A] . You are strongly encouraged to include a **justification to your answer**, either by referencing the appropriate section of your paper or providing a brief inline description. For example:

- Did you include the license to the code and datasets? [Yes] See Section ??.
- Did you include the license to the code and datasets? [No] The code and the data are proprietary.
- Did you include the license to the code and datasets? [N/A]

Please do not modify the questions and only use the provided macros for your answers. Note that the Checklist section does not count towards the page limit. In your paper, please delete this instructions block and only keep the Checklist section heading above along with the questions/answers below.

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 6.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Section 3 and 4.
   (b) Did you include complete proofs of all theoretical results? [Yes] See Section 3 and 4.
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] `https://anonymous.4open.science/r/MRI-2608/readme`
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Supplementary Materials.
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See Table 2 and 3.
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Supplementary Materials.
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 5.2 and Supplementary Materials.
   (b) Did you mention the license of the assets? [Yes] See Supplementary Materials.
   (c) Did you include any new assets either in the supplemental material or as a URL? [Yes] `https://anonymous.4open.science/r/MRI-2608/readme`
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]