
Invariant Causal Prediction for Block MDPs

Amy Zhang^{*123} Clare Lyle^{*4} Shagun Sodhani³ Angelos Filos⁴ Marta Kwiatkowska⁴ Joelle Pineau¹²³
Yarin Gal⁴ Doina Precup¹²⁵

Abstract

Generalization across environments is critical for the successful application of reinforcement learning algorithms to real-world challenges. In this paper, we consider the problem of learning abstractions that generalize in block MDPs, families of environments with a shared latent state space, and dynamics structure over that latent space, but varying observations. We leverage tools from causal inference to propose a method of invariant prediction to learn state abstractions that generalize to novel observations in the multi-environment setting. We prove that for certain classes of environments, this approach outputs with high probability a state abstraction corresponding to the causal feature set with respect to the return. We further provide more general bounds on model error and generalization error in the multi-environment setting in the process showing a connection between causal variable selection and the state abstraction framework for MDPs. We give empirical evidence that our methods work in both linear and nonlinear settings, attaining improved generalization over single- and multi-task baselines.

1. Introduction

The canonical reinforcement learning (RL) problem assumes an agent interacting with a single MDP with a fixed observation space and dynamics structure. This assumption is difficult to ensure in practice, where state spaces are often large and infeasible to explore entirely during training. However, there is often a latent structure to be leveraged to allow for good generalization. As an example, a robot’s sensors may be moved, or the lighting con-

ditions in a room may change, but the physical dynamics of the environment are still the same. These are examples of environment-specific characteristics that current RL algorithms often overfit to. In the worst case, some training environments may contain spurious correlations that will not be present at test time, causing catastrophic failures in generalization (Zhang et al., 2018a; Song et al., 2020). To develop algorithms that will be robust to these sorts of changes, we must consider problem settings that allow for multiple environments with a shared dynamics structure.

Recent works (Amit and Meir, 2018; Yin et al., 2019) have developed generalization bounds for the multi-task problem, but they depend on the number of tasks seen at training time, which can be prohibitively expensive given how sample inefficient RL is even in the single task regime. To obtain stronger generalization results, we propose to consider a problem which we refer to as ‘multi-environment’ RL: like multi-task RL, the agent seeks to maximize return on a set of environments, but only some of which it can be trained on. We make the assumption that there exists some latent causal structure that is shared among all of the environments, and that the sources of variability between environments do not affect the reward. This family of environments is called a *Block MDP* (Du et al., 2019), in which the observations may change, but the latent states, dynamics, and reward function are the same. A formal definition of this class of MDP will be presented in Section 3.

Though the setting we consider is a subset of the multi-task RL problem, we show that the added assumption of shared structure allows for much stronger generalization results than have been obtained by multi-task approaches. Naive application of generalization bounds to the multi-task reinforcement learning setting is very loose because the learner is typically given access to only a few tasks relative to the number of samples from each task. Indeed, Cobbe et al. (2018); Zhang et al. (2018b) find that agents trained using standard methods require many thousands of environments before ‘generalizing’ to new environments.

The main contribution of this paper is to use tools from *causal inference* to address generalization in the Block MDP setting, proposing a new method based on the *invariant causal prediction* literature. In certain linear function approximation settings, we demonstrate that this method

^{*}Equal contribution ¹McGill University ²Mila ³Facebook AI Research ⁴University of Oxford ⁵Deepmind. Correspondence to: Amy Zhang <amy.x.zhang@mail.mcgill.ca>, Clare Lyle <clare.lyle@univ.ox.ac.uk>.

will, with high probability, learn an optimal state abstraction that generalizes across all environments using many fewer training environments than would be necessary for standard PAC bounds. We replace this PAC requirement with requirements from causal inference on the *type* of environments seen at training time. We then draw a connection between bisimulation and the minimal causal set of variables found by our algorithm, providing bounds on the model error and sample complexity of the method. We further show that using analogous invariant prediction methods for the nonlinear function approximation setting can yield improved generalization performance over multi-task and single-task baselines. We relate this method to previous work on learning representations of MDPs (Gelada et al., 2019; Luo et al., 2019) and develop multi-task generalization bounds for such representations. Code is available at <https://github.com/facebookresearch/icp-block-mdp>.

2. Background

2.1. State Abstractions and Bisimulation

State abstractions have been studied as a way to distinguish relevant information from irrelevant one (Li et al., 2006) to create a more compact representation for easier decision making and planning. Bertsekas and Castanon (1989); Roy (2006) provide bounds for approximation errors for various aggregation methods, and Li et al. (2006) discuss the merits of *abstraction discovery* as a way to solve related MDPs.

Bisimulation relations are a type of state abstraction that offers a mathematically precise definition of what it means for two environments to ‘share the same structure’ (Larsen and Skou, 1989; Givan et al., 2003). We say that two states are bisimilar if they share the same expected reward and equivalent distributions over the next bisimilar states. For example, if a robot is given the task of washing the dishes in a kitchen, changing the wallpaper in the kitchen doesn’t change anything relevant to the task. One then could define a bisimulation relation that equates observations based on the locations and soil levels of dishes in the room and ignores the wallpaper. These relations can be used to simplify the state space for tasks like policy transfer (Castro and Precup, 2010), and are intimately tied to state abstraction. For example, the *model-irrelevance abstraction* described by Li et al. (2006) is precisely characterized as a bisimulation relation.

Definition 1 (Bisimulation Relations (Givan et al., 2003)). *Given an MDP \mathcal{M} , an equivalence relation B between states is a bisimulation relation if for all states $s_1, s_2 \in \mathcal{S}$ that are equivalent under B (i.e. $s_1 B s_2$), the following*

conditions hold for all actions $a \in \mathcal{A}$:

$$\begin{aligned} R(s_1, a) &= R(s_2, a) \\ \mathcal{P}(G|s_1, a) &= \mathcal{P}(G|s_2, a), \forall G \in \mathcal{S}/B \end{aligned}$$

Where \mathcal{S}/B denotes the partition of \mathcal{S} under the relation B , the set of all groups of equivalent states, and where $\mathcal{P}(G|s, a) = \sum_{s' \in G} \mathcal{P}(s'|s, a)$.

This definition was originally designed for the single MDP setting to find *bisimilar* states within an MDP. We are now trying to find bisimilar states across different MDPs, or different experimental conditions. One can intuitively think of this carrying over by imagining all experimental conditions i mapped to a single super-MDP with state space $\mathcal{S} = \cup_i \mathcal{S}_i$ where we give up the irreducibility assumption, i.e. we can no longer reach every state s_i from any other state s_j . Specifically, we say that two MDPs M_1 and M_2 are bisimilar if there exist bisimulation relations B_1 and B_2 such that M_1/B_1 is isomorphic to M_2/B_2 . *Bisimilar MDPs* are therefore MDPs which are behaviourally the same.

2.2. Causal Inference Using Invariant Prediction

Peters et al. (2016) introduced an algorithm, Invariant Causal Prediction (ICP), to find the *causal feature set*, the minimal set of features which are causal predictors of a target variable, by exploiting the fact that causal models have an invariance property (Pearl, 2009; Schölkopf et al., 2012). Arjovsky et al. (2019) extend this work by proposing invariant risk minimization (IRM, see Equation (1)), augmenting empirical risk minimization to learn a data representation free of spurious correlations. They assume there exists some partition of the training data \mathcal{X} into *experiments* $e \in \mathcal{E}$, and that the model’s predictions take the form $Y^e = \mathbf{w}^\top \phi(X^e)$. IRM aims to learn a representation ϕ for which the optimal linear classifier, \mathbf{w} , is invariant across e , where optimality is defined as minimizing the empirical risk R^e . We can then expect this representation and classifier to have low risk in new experiments e , which have the same causal structure as the training set.

$$\begin{aligned} \min_{\substack{\phi: \mathcal{X} \rightarrow \mathbb{R}^d \\ \mathbf{w} \in \mathbb{R}^d}} & \sum_{e \in \mathcal{E}} R^e(\mathbf{w}^\top \phi(X^e)) \\ \text{s.t. } & \mathbf{w} \in \arg \min_{\bar{\mathbf{w}} \in \mathbb{R}^d} R^e(\bar{\mathbf{w}}^\top \phi(X^e)) \quad \forall e \in \mathcal{E}. \end{aligned} \quad (1)$$

The IRM objective in Equation (1) can be interpreted as a constrained optimization problem, where the objective is to learn a set of features ϕ for which the optimal classifier in each environment is the same. Conditioned on the environments corresponding to different interventions on the data-generating process, this is hypothesized to yield features that only depend on variables that are causally related to the predicted value. Because the constrained opti-

mization problem is not generally feasible to optimize, Arjovsky et al. (2019) propose a penalized optimization problem with scheduled penalty term as a tractable alternative.

3. Problem Setup

We consider a family of environments $\mathcal{M}_{\mathcal{E}} = \{(\mathcal{X}_e, \mathcal{A}, \mathcal{R}_e, \mathcal{T}_e, \gamma) \mid e \in \mathcal{E}\}$, where \mathcal{E} is some index set. For simplicity of notation, we drop the subscript e when referring to the union over all environments \mathcal{E} . Our goal is to use a subset $\mathcal{E}_{\text{train}} \subset \mathcal{E}$ of these environments to learn a representation $\phi : \mathcal{X} \rightarrow \mathbb{R}^d$ which enables generalization of a learned policy to *every* environment. We denote the number of training environments as $N := |\mathcal{E}_{\text{train}}|$. We assume that the environments share some structure, and consider the different degrees to which this structure may be shared.

3.1. The Block MDP

Block MDPs (Du et al., 2019) are described by a tuple $\langle \mathcal{S}, \mathcal{A}, \mathcal{X}, p, q, R \rangle$ with a finite, unobservable state space \mathcal{S} , finite action space \mathcal{A} , and possibly infinite, but observable space \mathcal{X} . p denotes the latent transition distribution $p(s'|s, a)$ for $s, s' \in \mathcal{S}, a \in \mathcal{A}$, q is the (possibly stochastic) emission function that gives the observations from the latent state $q(x|s)$ for $x \in \mathcal{X}, s \in \mathcal{S}$, and R the reward function. A graphical model of the interactions between the various variables can be found in Figure 1.

Assumption 1 (Block structure (Du et al., 2019)). *Each observation x uniquely determines its generating state s . That is, the observation space \mathcal{X} can be partitioned into disjoint blocks \mathcal{X}_s , each containing the support of the conditional distribution $q(\cdot|s)$.*

This assumption gives us the Markov property in \mathcal{X} . We translate the block MDP to our multi-environment setting as follows. If a family of environments $\mathcal{M}_{\mathcal{E}}$ satisfies the block MDP assumption, then each $e \in \mathcal{E}$ corresponds to an emission function q_e , with $\mathcal{S}, \mathcal{A}, \mathcal{X}$ and p shared for all $M \in \mathcal{M}_{\mathcal{E}}$. We will move the potential randomness from q_e into an auxiliary variable $\eta \in \Omega$, and write $q_e(\eta, s)$. Further, we require that if $\text{range}(q_e(\cdot, s)) \cap \text{range}(q_{e'}(\cdot, s')) \neq \emptyset$, then $s = s'$. The objective is to learn a useful state abstraction to promote generalization across the different emission functions q_e , given that only a subset is provided for training. Song et al. (2020) also describes a similar POMDP setting where there is an additional observation function, but assume information can be lost. We note that this problem can be made arbitrarily difficult if each q_e has a disjoint range, but will focus on settings where the q_e overlap in structured ways – for example, where q_e is the concatenation of the noise and state variables: $q_e(\eta, s) = s \oplus f(\eta)$.

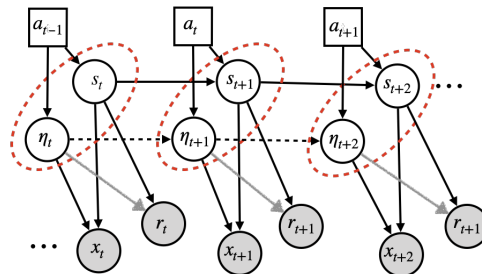


Figure 1. Graphical model of a block MDP with stochastic, correlated observations, with an IRM goal to extract s from the sequence of observations, and discard the spurious noise η . Red dashed ovals indicate the entire tangled latent state at each timestep. Black dashed lines and grey lines indicate two additional tiers of difficulty to consider. (Best viewed in color.)

3.2. Relaxations

Spurious correlations. Our initial presentation of the block MDP assumes that the noise variable η is sampled randomly at every time step, which prevents multi-timestep correlations (Figure 1 in black, solid lines). We therefore also consider a more realistic *relaxed block MDP*, where spurious variables may have different transition dynamics across the different environments so long as these correlations do not affect the expected reward (Figure 1, now including black dashed lines). This is equivalent to augmenting each Block MDP in our family with a noise variable η_e , such that the observation $x = (q_e(\eta_e, s))$, and

$$p(x'|x, a) = p(q^{-1}(x')|s, a)p_e(\eta'_e|s, \eta_e).$$

We note that this section still satisfies Assumption 1.

Realizability. Though our analysis will require Assumption 1, we claim that this is a reasonable requirement as it makes the learning problem realizable. Relaxing Assumption 1 means that the value function learning problem may become ill-posed, as the same observation can map to entirely different states in the latent MDP with different values, making our environment partially observable (a POMDP, Figure 1 with grey lines). We provide a lower bound on the value approximation error attainable in this setting in the appendix (Proposition 2).

3.3. Assumptions on causal structure

State abstraction and causal inference both aim to eliminate spurious features in a learning algorithm’s input. However, these two approaches are applied to drastically different types of problems. Though we demonstrate that causal inference methods can be applied to reinforcement learning, this will require some assumption on how causal mechanisms are observed. Definitions of the notation used in this section are deferred to the appendix, though they are standard in the causal inference literature.

The key assumption we make is that the variables in the environment state at time t can only affect the values of the state at time $t + 1$, and can only affect the reward at time t . This assumption enables us to consider the state and action at time t as the only candidate for causal parents of the state at time $t + 1$ and of the reward at time t . This assumption is crucial to the Markov behaviour of the Markov decision process. We refer the reader to Figure 2 to demonstrate how causal graphical models can be translated to this setting.

Assumption 2 (Temporal Causal Mechanisms). *Let x^1 and x^2 be components of the observation x . Then when no intervention is performed on the environment, we have the following independence.*

$$X_{t+1}^1 \perp X_{t+1}^2 | x_t \quad (2)$$

Assumption 3 (Environment Interventions). *Let $\mathcal{X} = X_1 \times \dots \times X_n$, and $\mathcal{S} = X_{i_1} \times \dots \times X_{i_k}$. Each environment $e \in \mathcal{E}$ corresponds to a do- (Pearl, 2009) or soft (Eberhardt and Scheines, 2007) intervention on a single variable x_i in the observation space.*

This assumption allows us to use tools from causal inference to identify candidate model-irrelevance state abstractions that may hold across an entire family of MDPs, rather than only the ones observed, based on using the state at one timestep to predict the values at the next timestep. In the setting of Assumption 3, we can reconstruct the block MDP emission function q by concatenating the spurious variables from $\mathcal{X} \setminus \mathcal{S}$ to \mathcal{S} . We discuss some constraints on interventions necessary to satisfy the block MDP assumption in the appendix.

4. Connecting State Abstractions to Causal Feature Sets

Invariant Causal Prediction (Peters et al., 2016) aims to identify a set S of causal variables such that a linear predictor with support on S will attain consistent performance over all environments. In other words, ICP removes irrelevant variables from the input, just as state abstractions remove irrelevant information from the environment’s observations. An attractive property of the block MDP setting is that it is easy to show that there does exist a model-irrelevance state abstraction ϕ for all MDPs in $\mathcal{M}_{\mathcal{E}}$ – namely, the function mapping each observation x to its generating latent state $\phi(x) = q^{-1}(x)$. The formalization and proof of this statement are deferred to the appendix (see Theorem 4).

We consider whether, under Assumptions 1-3, such a state abstraction can be obtained by ICP. Intuitively, one would then expect that the *causal variables* should have nice properties as a state abstraction. The following result confirms this to be the case; a state abstraction that selects the set

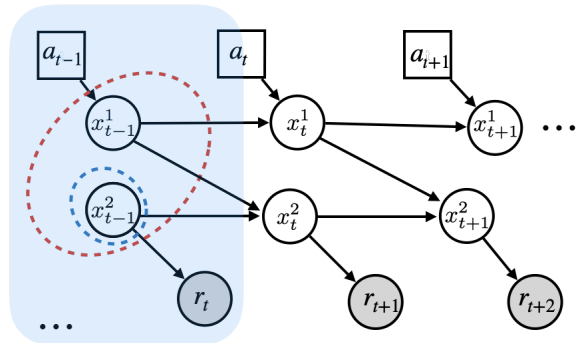


Figure 2. Graphical causal models with temporal dependence – note that while x^2 (circled in blue) is the only causal parent of the reward, because its next-timestep distribution depends on x^1 , a model-irrelevance state abstraction must include both variables. Shaded in blue: the graphical causal model of an MDP with states $s = (x^1, x^2)$ when ignoring timesteps.

of causal variables from the observation space of a block MDP will be a model-irrelevance abstraction for every environment $e \in \mathcal{E}$.

Theorem 1. *Consider a family of MDPs $M_{\mathcal{E}} = \{(\mathcal{X}, A, R, P_e, \gamma) | e \in \mathcal{E}\}$, with $\mathcal{X} = \mathbb{R}^k$. Let $M_{\mathcal{E}}$ satisfy Assumptions 1-3. Let $S_R \subseteq \{1, \dots, k\}$ be the set of variables such that the reward $R(x, a)$ is a function only of $[x]_{S_R}$ (x restricted to the indices in S_R). Then let $S = \text{AN}(R)$ denote the ancestors of S_R in the (fully observable) causal graph corresponding to the transition dynamics of $M_{\mathcal{E}}$. Then the state abstraction $\phi_S(x) = [x]_S$ is a model-irrelevance abstraction for every $e \in \mathcal{E}$.*

An important detail in the previous result is the model irrelevance state abstraction incorporates not just the parents of the reward, but also its ancestors. This is because in RL, we seek to model *return* rather than solely rewards, which requires a state abstraction that can capture multi-timestep interactions. We provide an illustration of this in Figure 2. As a concrete example, we note that in the popular benchmark CartPole, only position x and angle θ are necessary to predict the reward. However, predicting the return requires θ and \dot{x} , their respective velocities.

Learning a minimal ϕ in the setting of Theorem 1 using a single training environment may not always be possible. However, applying invariant causal prediction methods in the multi-environment setting will yield the minimal causal set of variables when the training environment interventions satisfy certain conditions necessary for the identifiability of the causal variables (Peters et al., 2016).

5. Block MDP Generalization Bounds

We continue to relax the assumptions needed to learn a causal representation and look to the nonlinear setting. As

a reminder, the goal of this work is to produce representations that will generalize from the training environments to a novel test environment. However, normal PAC generalization bounds require a much larger number of environments than one could expect to obtain in the reinforcement learning setting. The appeal of an invariant representation is that it may allow for theoretical guarantees on learning the right state abstraction with many fewer training environments, as discussed by Peters et al. (2016). If the learned state abstraction is close to capturing the true base MDP, then the model error in the test environment can be bounded by a function of the distance of the test environment’s abstract state distribution to the training environments’. Though the requirements given in the following Theorem 2 are difficult to guarantee in practice, the result will hold for any arbitrary learned state abstraction.

Theorem 2 (Model error bound). *Let M_1 and M_2 be two environments satisfying Assumption 1, and let $\phi : \mathcal{X} \rightarrow \mathcal{Z}$ be a model-irrelevance abstraction for M_1 and M_2 . Let the union of the environments’ state transition functions T_1 and T_2 be L -Lipschitz with respect to the state embedding $\phi(X)$, and T be an arbitrary learned transition function defined on \mathcal{Z} . Setting the expected error of T on M_1 as $\mathbb{E}_{x \sim \pi(M_1)}[\|T(\phi(x)) - \phi(T_1(x))\|] = \delta$, we have the following bound on the error of T in M_2*

$$\mathbb{E}_{x \sim M'}[\|T(\phi(x)) - \phi(T_2(x))\|] \leq \delta + 2LW_1(\pi_{\phi(M_1)}, \pi_{\phi(M_2)}). \quad (3)$$

Proof provided in Appendix B.

Instead of assuming access to a bisimilar MDP M' , we can provide discrepancy bounds for an MDP \bar{M} produced by a learned state representation $\phi(x)$, dynamics function f_s , and reward function R using the distance in dynamics J_D^∞ and reward J_R^∞ of \bar{M} to the underlying MDP M . We first define these distances,

$$\begin{aligned} J_R^\infty &:= \sup_{x \in \mathcal{X}, a \in \mathcal{A}} |R(\phi(x), a, \phi(x')) - r(x, a)| \\ J_D^\infty &:= \sup_{x \in \mathcal{X}, a \in \mathcal{A}} W_1(f_s(\phi(x), a), \phi P(x, a)). \end{aligned} \quad (4)$$

Theorem 3. *Let M be a block MDP and \bar{M} the learned invariant MDP with a mapping $\phi : \mathcal{X} \mapsto \mathcal{Z}$. For any L -Lipschitz valued policy π the value difference of that policy is bounded by*

$$|Q^\pi(x, a) - \bar{Q}^\pi(\phi(x), a)| \leq \frac{J_R^\infty + \gamma L J_D^\infty}{1 - \gamma}, \quad (5)$$

where Q^π is the value function for π in M and \bar{Q}^π is the value function for π in \bar{M} .

Proof provided in Appendix B. This gives us a bound on generalization performance that depends on the supremum of the dynamics and reward errors, which correspondingly

is a regression problem that will depend on $\sum_{e \in \mathcal{E}} n_e$, the number of samples we have in aggregate over all training environments rather than the number of training environments, $|\mathcal{E}|$. Recent generalization bounds for deep neural networks using Rademacher complexity (Bartlett et al., 2017a; Arora et al., 2018) scale with a factor of $\frac{1}{\sqrt{n}}$ where n is the number of samples. We can use $n := \sum_{e \in \mathcal{E}} n_e$ for our setting, getting generalization bounds for the block MDP setting that scale with the number of samples in aggregate over all environments, an improvement over previous multi-task bounds that depend on $|\mathcal{E}|$.

6. Methods

Given these theoretical results, we propose two methods to learn invariant representations in the block MDP setting. Both methods take inspiration from invariant causal prediction, with the first being the direct application of linear ICP to select the causal variables in the state in the setting where variables are given. This corresponds to direct feature selection, which with high probability returns the minimal causal feature set. The second method is a gradient-based approach akin to the IRM objective, with no assumption of a linear causal relationship and a learned causal invariant representation. Like the IRM goal (Equation (1)), we aim to learn an invariant state abstraction from stochastic observations across different interventions i , and impose an additional invariance constraint.

6.1. Variable Selection for Linear Predictors

The following algorithm (Algorithm 1) returns a model-irrelevance state abstraction. We require the presence of a replay buffer \mathcal{D} , in which transitions are stored and tagged with the environment from which they came. The algorithm then applies ICP to find all causal ancestors of the reward iteratively. This approach has the benefit of inheriting many nice properties from ICP – under suitable identifiability conditions, it will return the exact causal variable set to a specified degree of confidence.

It also inherits inconvenient properties: the ICP algorithm is exponential in the number of variables, and so this method is not efficient for high-dimensional observation spaces. We are also restricted to considering linear relationships of the observation to the reward and next state. Further, because we take the union over iterative applications of ICP, the confidence parameter α used in each call must be adjusted accordingly. Given n observation variables, we give a conservative bound of $\frac{\alpha}{n}$.

While ICP is limited to linear relationships, Heinze-Deml et al. (2018) extends the algorithm to the nonlinear setting, and therefore can be substituted into Algorithm 1 in the same way. However, it still is a feature selection method

Algorithm 1 Linear MISA: Model-irrelevance State Abstractions

Result: $S \subset \{1, \dots, k\}$, the causal state variables
Input: α , a confidence parameter, \mathcal{D} , an replay buffer with observations \mathcal{X} .
 $S \leftarrow \emptyset$
 stack $\leftarrow r$
while stack is not empty **do**
 $v = \text{stack.pop}()$
 if $v \notin S$ **then**
 $S' \leftarrow \text{ICP}(v, \mathcal{D}, \frac{\alpha}{\dim(\mathcal{X})})$
 $S \leftarrow S \cup S'$
 stack.push(S')
 end
end

that assumes the relevant features are given, and therefore does not scale to pixel observations.

6.2. Learning a Model-irrelevance State Abstraction

We design an objective to learn a dynamics preserving state abstraction \mathcal{Z} , or model-irrelevance abstraction (Li et al., 2006), where the similarity of the model is bounded by the model error in the environment setting shown in Figure 1. This requires disentangling the state space into a minimal representation that causes reward $s_t := \phi(x_t)$ and everything else $\eta_t := \varphi(x_t)$. Our algorithm proceeds as follows.

We assume the existence of an invariant state embedding, whose mapping function we denote by $\phi : \mathcal{X} \mapsto \mathcal{Z}$. We also assume an invariant dynamics model $f_s : \mathcal{A} \times \mathcal{Z} \mapsto \mathcal{Z}$, a task-specific dynamics model $f_\eta : \mathcal{A} \times \mathcal{H} \mapsto \mathcal{H}$, and an invariant reward model $r : \mathcal{Z} \times \mathcal{A} \times \mathcal{Z} \mapsto \mathbb{R}$ in the embedding space. To incorporate a meaningful objective and ground the learned representation, we need a decoder $\phi^{-1} : \mathcal{Z} \times \mathcal{H} \mapsto \mathcal{X}$. We assume $N > 1$ training environments are given. The overall dynamics and reward objectives become

$$J_D(\phi, \psi, f_s, f_\eta) = \sum_i \mathbb{E}_{\pi_{b_i}} [(\phi^{-1}(f_s(a, \phi(x_i))), f_\eta(a, \psi(x_i))) - x'_i)^2],$$

$$J_R(\phi, R) = \sum_i \mathbb{E}_{\pi_{b_i}} [(R(\phi(x_i), a, \phi(x'_i)) - r'_i)^2],$$

under data collected from behavioral policies π_{b_i} for each experimental setting.

Of course, this does not guarantee that the representation learned by ϕ will be minimal, so we incorporate additional regularization as an incentive. We train a task classifier on the shared latent representation $C : \mathcal{Z} \mapsto [0, 1]^N$ with cross-entropy loss and employ an adversarial loss (Tzeng

Algorithm 2 Nonlinear Model-irrelevance State Abstraction (MISA) Learning

Result: ϕ , an invariant state encoder
 $\pi \leftarrow \pi_0$
 $\phi, f_s \leftarrow \phi_0, f_{s,0}$
 $\psi^e, f_\eta^e \leftarrow \psi_0^e, f_{\eta,0}^e$ for $e \in \mathcal{E}$
 $\mathcal{D}_e \leftarrow \emptyset$ for $e \in \mathcal{E}$
while forever do
 for $e \in \mathcal{E}$ **do**
 $a \leftarrow \pi(x_e)$
 $x'_e, r \leftarrow \text{step}(x_e, a)$
 store(x_e, a, r, x'_e)
 end
 for $e \in \mathcal{E}$ **do**
 Sample batch X_e from \mathcal{D}_e
 $f_\eta^e, \psi^e \leftarrow \nabla_{f_\eta^e, \psi^e} [J_D(X_e)]$
 end
 $f_s, \phi, r \leftarrow \sum_{X_e} \nabla_{f_s, \phi} [J_{\text{ALL}}(X_e)]$
 $C \leftarrow \nabla_C \text{CE-LOSS}(C(\{\phi(x_e)\}_{e \in \mathcal{E}}), \{e\}_{e \in \mathcal{E}})$
end

et al., 2017) on ϕ to maximize the entropy of the classifier output to ensure task specific information is not passing through to \mathcal{Z} .

This gives us a final objective

$$J_{\text{ALL}}(\phi, \psi, f_s, f_\eta, r) = J_D(\phi, \psi, f_s, f_\eta) + \alpha_R J_R(\phi, r) - \alpha_C H(C(\phi)), \quad (6)$$

where α_R and α_C are hyperparameters and H denotes entropy (Algorithm 2).

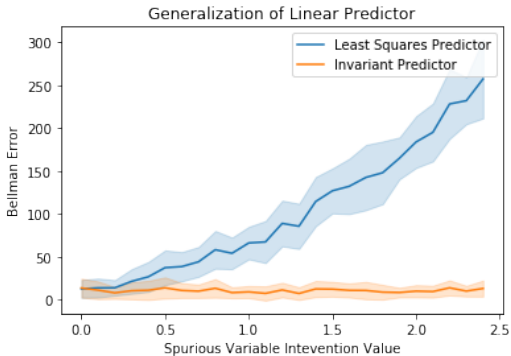
7. Results

We evaluate both linear and non-linear versions of MISA, in corresponding Block MDP settings with both linear and non-linear dynamics. First, we examine model error in environments with low-dimensional (Section 7.1.1) and high-dimensional (Section 7.1.2) observations and demonstrate the ability for MISA to zero-shot generalize to unseen environments. We next look to imitation learning in a rich observation setting (Section 7.2) and show non-linear MISA generalize to new camera angles. Finally, we explore end-to-end reinforcement learning in the low-dimensional observation setting with correlated noise (Section 7.3) and again show generalization capabilities where single task and multi-task methods fail.

7.1. Model Learning

7.1.1. LINEAR SETTING

We first evaluate the linear MISA algorithm in [Algorithm 1](#). To empirically evaluate whether eliminating spurious variables from a representation is necessary to guarantee generalization, we consider a simple family of MDPs with state space $\mathcal{X} = \{(x_1, x_2, x_3)\}$, with a transition dynamics structure such that $x_1^{t+1} = x_1^t + \epsilon_1^e$, $x_2^{t+1} = x_1^t + \epsilon_2^e$, and $x_3^{t+1} = x_2^t + \epsilon_3^e$. We train on 3 environments with soft interventions on each noise variable. We then run the linear MISA algorithm on batch data from these 3 environments to get a state abstraction $\phi(x) = \{x_1, x_2\}$, and then train 2 linear predictors on $\phi(x)$ and x . We then evaluate the *generalization performance* for novel environments that correspond to different hard interventions on the value of the x_3 variable. We observe that the predictor trained on $\phi(x)$ attains zero generalization error because it zeros out x_3 automatically. However, any nonzero weight on x_3 in the least-squares predictor will lead to arbitrarily large generalization error, which is precisely what we observe in [Figure 3](#).

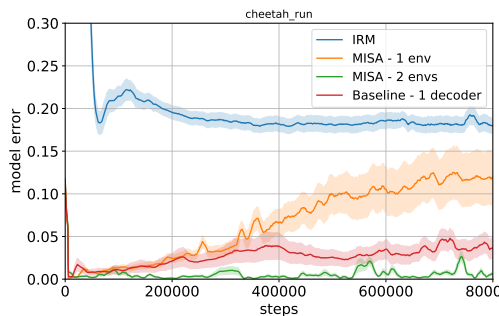


[Figure 3](#). The presence of spurious *uncorrelated* variables in the state can still lead to poor generalization of linear function approximation methods. Invariant Causal Prediction methods can eliminate these spurious variables altogether.

7.1.2. RICH OBSERVATION SETTING

We next test the gradient-based MISA method ([Algorithm 2](#)) in a setting with nonlinear dynamics and rich observations. Instead of having access to observation variables and selecting the minimal causal feature set, we are tasked with learning the invariant causal representation. We randomly initialize the background color of two train environments from Deepmind Control ([Tassa et al., 2018](#)) from range $[0, 255]$. We also randomly initialize another two backgrounds for evaluation. The orange line in [Figure 4](#) shows performance on the evaluation environments in comparison to three baselines. In the first, we only train on a single environment and test on another with our method,

(MISA - 1 env). Without more than a single experiment to observe at training time, there is no way to disentangle what is causal in terms of dynamics, and what is not. In the second baseline, we combine data from the two environments and train a model over all data (Baseline - 1 decoder). The third is another invariance-based method which uses a gradient penalty, IRM ([Arjovsky et al., 2019](#)). In the second case the error is tempered by seeing variance in the two environments at training time, but it is not as effective as MISA with two environments at disentangling what is invariant, and therefore causal with respect to dynamics, and what is not. With IRM, the loss starts much higher but very slowly decreases, and we find it is very brittle to tune in practice. Implementation details found in [Appendix C.1](#).



[Figure 4](#). Model error on evaluation environments on Cheetah Run from Deepmind Control. 10 seeds, with one standard error shaded.

7.2. Imitation Learning

In this setup, we first train an expert policy using the proprioceptive state of Cheetah Run from ([Tassa et al., 2018](#)). We then use this policy to collect a dataset for imitation learning in each of two training environments. When rendering these high dimensional images, we alter the camera angles in the different environments ([Figure 5](#)). We report the generalization performance as the test error when predicting actions in [Figure 6](#). While we see test error does increase with our method, MISA, the error growth is significantly slower compared to single task and multi-task baselines. We note that after significant tuning, the IRM baseline performs better than the proposed baseline. One reason for the better performance of the IRM baseline in this setup (and not other setups) could be that this setup uses a stationary data distribution (because of the use of a fixed replay buffer).

7.3. Reinforcement Learning

We go back to the proprioceptive state in the `cartpole_swingup` environment in Deepmind Control ([Tassa et al., 2018](#)) to show that we can learn MISA



Figure 5. The Cheetah Run environment from Deepmind Control with different camera angles. The first two images are from the training environments and the last image is from evaluation environment.

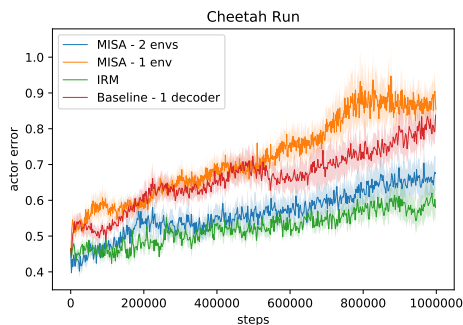


Figure 6. Actor error on evaluation environments on Cheetah Run from Deepmind Control. 10 seeds, with one standard error shaded.

while training a policy. We use Soft Actor Critic (Haarnoja et al., 2018) with an additional linear encoder, and add spurious correlated dimensions which are a multiplicative factor of the original state space. We also add an additional environment identifier to the observation. This multiplicative factor varies across environments, and we train on two environments with $1\times$ and $2\times$, and test on $3\times$. Like Arjovsky et al. (2019), we also incorporate noise on the causal state to make the task harder, specifically Gaussian noise $\mathcal{N}(0, 0.01)$ to the true state dimension. This incentivizes the agent to attend to the spuriously correlated dimension instead, which has no noise. In Figure 7 we see the generalization gap drastically improve with our method in comparison to training SAC with data over all environments in aggregate and with IRM (Arjovsky et al., 2019) implemented on the critic loss. Implementation details and more information about Soft Actor Critic can be found in Appendix C.2. Additional plots with the hyperparameter sweep done to find a good penalty weight for IRM can also be found Appendix C.3.

8. Related Work

8.1. Prior Work on Generalization Bounds

Generalization bounds provide guarantees on the test set error attained by an algorithm. Most of these bounds are probabilistic and targeted at the supervised setting, falling into the PAC (Probably Approximately Correct) frame-

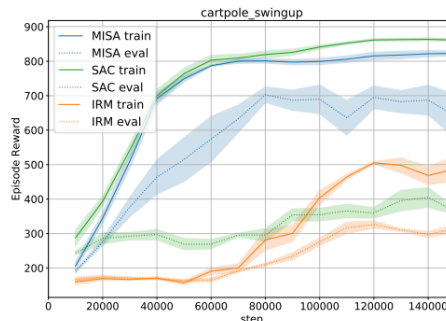


Figure 7. Generalization gap in SAC performance with 2 training environments on `cartpole_swingup` from DMC. Evaluated with 10 seeds, standard error shaded.

work. PAC bounds give probabilistic guarantees on a model’s true error as a function of its train set error and the complexity of the function class encoded by the model. Many measures of hypothesis class complexity exist: the Vapnik-Chervonenkis (VC) dimension (Vapnik and Chervonenkis, 1971), the Lipschitz constant, and classification margin of a neural network (Bartlett et al., 2017b), and second-order properties of the loss landscape (Neyshabur et al., 2019) are just a few of many.

Analogous techniques can be applied to Bayesian methods, giving rise to PAC-Bayes bounds (McAllester, 1999). This family of bounds can be optimized to yield non-vacuous bounds on the test error of over-parametrized neural networks (Dziugaite and Roy, 2017), and have demonstrated strong empirical correlation with model generalization (Jiang* et al., 2020). More recently, Amit and Meir (2018); Yin et al. (2019) introduce a PAC-Bayes bound for the multi-task setting dependent on the number of tasks seen at training time.

Strehl et al. (2006) extend PAC framework to reinforcement learning, defining a new class of bounds called PAC-MDP. An algorithm is PAC-MDP if for any ϵ and δ , the sample complexity of the algorithm is less than some polynomial in $(S, A, 1/\epsilon, 1/\delta, 1/(1-\gamma))$ with probability at least $1-\delta$. The authors provide a PAC-MDP algorithm for model-free Q-learning. Lattimore and Hutter (2012) offers lower and upper bounds on the sample complexity of learning near-optimal behavior in MDPs by modifying the Upper Confidence RL (UCRL) algorithm (Jaksch et al., 2010).

8.2. Multi-Task Reinforcement Learning

Teh et al. (2017); Borsa et al. (2016) handle multi-task reinforcement learning with a shared “distilled” policy (Teh et al., 2017) and shared state-action representation (Borsa et al., 2016) to capture common or invariant behavior

across all tasks. No assumptions are made about how these tasks relate to each other other than a shared state and action space. D’Eramo et al. (2020) show the benefits of learning a shared representation in multi-task settings with an approximate value iteration bound and Brunskill and Li (2013) also demonstrate a PAC-MDP algorithm with improved sample efficiency bounds through transfer across similar tasks. Again, none of these works look to the multi-environment setting to explicitly leverage environment structure. Barreto et al. (2017) exploit successor features for transfer, making the assumption that the dynamics across tasks are the same, but the reward changes. However, they do not handle the setting where states are latent, and observations change.

9. Discussion

This work has demonstrated that given certain assumptions, we can use causal inference methods in reinforcement learning to learn an invariant causal representation that generalizes across environments with a shared causal structure. We have provided a framework for defining families of environments, and methods, for both the low dimensional linear value function approximation setting and the deep RL setting, which leverage invariant prediction to extract a causal representation of the state. We have further provided error bounds and identifiability results for these representations. We see this paper as a first step towards the more significant problem of learning useful representations for generalization across a broader class of environments. Some examples of potential applications include third-person imitation learning, sim2real transfer, and, related to sim2real transfer, the use of privileged information in one task (the simulation) as grounding and generalization to new observation spaces (Salter et al., 2019).

10. Acknowledgements

MK has received funding from the European Research Council (ERC) under the European Union’s Horizon 2020 research and innovation programme (grant agreement No. 834115). The authors would also like to thank Marlos Machado for helpful feedback in the writing process.

References

Amit, R. and Meir, R. (2018). Meta-learning by adjusting priors based on extended PAC-Bayes theory. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 205–214, Stockholmsmässan, Stockholm Sweden. PMLR.

- Arjovsky, M., Bottou, L., Gulrajani, I., and Lopez-Paz, D. (2019). Invariant Risk Minimization. *arXiv e-prints*.
- Arora, S., Ge, R., Neyshabur, B., and Zhang, Y. (2018). Stronger generalization bounds for deep nets via a compression approach. In Krause, A. and Dy, J., editors, *35th International Conference on Machine Learning, ICML 2018*, 35th International Conference on Machine Learning, ICML 2018, pages 390–418. International Machine Learning Society (IMLS).
- Ba, J. L., Kiros, J. R., and Hinton, G. E. (2016). Layer normalization. *arXiv e-prints*.
- Barreto, A., Dabney, W., Munos, R., Hunt, J. J., Schaul, T., van Hasselt, H. P., and Silver, D. (2017). Successor features for transfer in reinforcement learning. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 4055–4065. Curran Associates, Inc.
- Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. (2017a). Spectrally-normalized margin bounds for neural networks. In Guyon, I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 6240–6249. Curran Associates, Inc.
- Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. (2017b). Spectrally-normalized margin bounds for neural networks. In *Advances in Neural Information Processing Systems*, pages 6240–6249.
- Bertsekas, D. and Castanon, D. (1989). Adaptive aggregation for infinite horizon dynamic programming. *Automatic Control, IEEE Transactions on*, 34:589 – 598.
- Borsa, D., Graepel, T., and Shawe-Taylor, J. (2016). Learning shared representations in multi-task reinforcement learning. *CoRR*, abs/1603.02041.
- Brunskill, E. and Li, L. (2013). Sample complexity of multi-task reinforcement learning. *Uncertainty in Artificial Intelligence - Proceedings of the 29th Conference, UAI 2013*.
- Castro, P. S. and Precup, D. (2010). Using bisimulation for policy transfer in mdps. In *Twenty-Fourth AAAI Conference on Artificial Intelligence*.
- Cobbe, K., Klimov, O., Hesse, C., Kim, T., and Schulman, J. (2018). Quantifying generalization in reinforcement learning. *CoRR*, abs/1812.02341.
- D’Eramo, C., Tateo, D., Bonarini, A., Restelli, M., and Peters, J. (2020). Sharing knowledge in multi-task deep reinforcement learning. In *International Conference on Learning Representations*.

- Du, S. S., Krishnamurthy, A., Jiang, N., Agarwal, A., Dudík, M., and Langford, J. (2019). Provably efficient RL with rich observations via latent state decoding. *CoRR*, abs/1901.09018.
- Dziugaite, G. K. and Roy, D. M. (2017). Computing nonvacuous generalization bounds for deep (stochastic) neural networks with many more parameters than training data.
- Eberhardt, F. and Scheines, R. (2007). Interventions and causal inference. *Philosophy of Science*, 74(5):981–995.
- Gelada, C., Kumar, S., Buckman, J., Nachum, O., and Bellemare, M. G. (2019). DeepMDP: Learning continuous latent space models for representation learning. In Chaudhuri, K. and Salakhutdinov, R., editors, *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pages 2170–2179, Long Beach, California, USA. PMLR.
- Givan, R., Dean, T. L., and Greig, M. (2003). Equivalence notions and model minimization in markov decision processes. *Artif. Intell.*, 147:163–223.
- Haarnoja, T., Zhou, A., Abbeel, P., and Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. In Dy, J. and Krause, A., editors, *Proceedings of the 35th International Conference on Machine Learning*, volume 80 of *Proceedings of Machine Learning Research*, pages 1861–1870, Stockholm, Sweden. PMLR.
- Heinze-Deml, C., Peters, J., and Meinshausen, N. (2018). Invariant causal prediction for nonlinear models. *Journal of Causal Inference*, 6(2).
- Jaksch, T., Ortner, R., and Auer, P. (2010). Near-optimal regret bounds for reinforcement learning. *J. Mach. Learn. Res.*, 11:1563–1600.
- Jiang*, Y., Neyshabur*, B., Krishnan, D., Mobahi, H., and Bengio, S. (2020). Fantastic generalization measures and where to find them. In *International Conference on Learning Representations*.
- Larsen, K. G. and Skou, A. (1989). Bisimulation through probabilistic testing (preliminary report). In *Proceedings of the 16th ACM SIGPLAN-SIGACT Symposium on Principles of Programming Languages*, POPL '89, page 344–352, New York, NY, USA. Association for Computing Machinery.
- Lattimore, T. and Hutter, M. (2012). Pac bounds for discounted mdps. In *International Conference on Algorithmic Learning Theory*, pages 320–334. Springer.
- Li, L., Walsh, T., and Littman, M. (2006). Towards a unified theory of state abstraction for mdps.
- Luo, Y., Xu, H., Li, Y., Tian, Y., Darrell, T., and Ma, T. (2019). Algorithmic framework for model-based deep reinforcement learning with theoretical guarantees. In *International Conference on Learning Representations*.
- McAllester, D. A. (1999). Pac-bayesian model averaging. In *COLT*, volume 99, pages 164–170. Citeseer.
- Neyshabur, B., Li, Z., Bhojanapalli, S., LeCun, Y., and Srebro, N. (2019). The role of over-parametrization in generalization of neural networks. In *International Conference on Learning Representations*.
- Pearl, J. (2009). *Causality: Models, Reasoning and Inference*. Cambridge University Press, New York, NY, USA, 2nd edition.
- Peters, J., Bühlmann, P., and Meinshausen, N. (2016). Causal inference using invariant prediction: identification and confidence intervals. *Journal of the Royal Statistical Society, Series B (with discussion)*, 78(5):947–1012.
- Roy, B. V. (2006). Performance loss bounds for approximate value iteration with state aggregation. *Math. Oper. Res.*, 31(2):234–244.
- Salter, S., Rao, D., Wulfmeier, M., Hadsell, R., and Posner, I. (2019). Attention privileged reinforcement learning for domain transfer.
- Schölkopf, B., Janzing, D., Peters, J., Sgouritsa, E., Zhang, K., and Mooij, J. (2012). On causal and anticausal learning. In *Proceedings of the 29th International Conference on Machine Learning*, ICML'12, page 459–466, Madison, WI, USA. Omnipress.
- Song, X., Jiang, Y., Tu, S., Du, Y., and Neyshabur, B. (2020). Observational overfitting in reinforcement learning. In *International Conference on Learning Representations*.
- Strehl, A. L., Li, L., Wiewiora, E., Langford, J., and Littman, M. L. (2006). Pac model-free reinforcement learning. In *Proceedings of the 23rd international conference on Machine learning*, pages 881–888. ACM.
- Tassa, Y., Doron, Y., Muldal, A., Erez, T., Li, Y., de Las Casas, D., Budden, D., Abdolmaleki, A., Merel, J., Lefrancq, A., Lillicrap, T., and Riedmiller, M. (2018). DeepMind control suite. Technical report, DeepMind.
- Teh, Y., Bapst, V., Czarnecki, W. M., Quan, J., Kirkpatrick, J., Hadsell, R., Heess, N., and Pascanu, R. (2017). Distral: Robust multitask reinforcement learning. In Guyon,

- I., Luxburg, U. V., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., and Garnett, R., editors, *Advances in Neural Information Processing Systems 30*, pages 4496–4506. Curran Associates, Inc.
- Tzeng, E., Hoffman, J., Saenko, K., and Darrell, T. (2017). Adversarial discriminative domain adaptation. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2962–2971, Los Alamitos, CA, USA. IEEE Computer Society.
- Vapnik, V. N. and Chervonenkis, A. Y. (1971). On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications*, 16(2):264–280.
- Yarats, D. and Kostrikov, I. (2020). Soft actor-critic (sac) implementation in pytorch. https://github.com/denisyarats/pytorch_sac.
- Yin, M., Tucker, G., Zhou, M., Levine, S., and Finn, C. (2019). Meta-learning without memorization. *arXiv preprint arXiv:1912.03820*.
- Zhang, A., Wu, Y., and Pineau, J. (2018a). Natural environment benchmarks for reinforcement learning. *CoRR*, abs/1811.06032.
- Zhang, C., Vinyals, O., Munos, R., and Bengio, S. (2018b). A study on overfitting in deep reinforcement learning. *CoRR*, abs/1804.06893.

A. Notation

We provide a summary of key notation used throughout the paper here.

$\mathbf{PA}_{\mathcal{G}}(X)$: the parents of node X in the causal graph \mathcal{G} . When \mathcal{G} is clear from the setting, abbreviate this notation to $\mathbf{PA}(X)$.

$\mathbf{AN}_{\mathcal{G}}(X)$: the ancestors of node X in \mathcal{G} (again, \mathcal{G} omitted when unambiguous).

$[x]_S : [x_{i_1}, \dots, x_{i_k} | i_j \in S]$

π_M : the stationary distribution given by some fixed policy in an MDP M .

q : the emission function of a block MDP.

\mathcal{E} : a set of environments.

B. Proofs

Technical notes and assumptions. In order for the block MDP assumption to be satisfied, we will require that the interventions defining each environment only occur outside of the causal ancestors of the reward. Otherwise, the different environments will have different latent state dynamics, which violates our assumption that the environments are obtained by a noisy emission function from the latent state space \mathcal{S} . Although ICP will still find the correct causal variables in this setting, this state abstraction will no longer be a model irrelevance state abstraction over the union of the environments.

Theorem 1. Consider a family of MDPs $M_{\mathcal{E}} = \{(\mathcal{X}, A, R, P_e, \gamma) | e \in \mathcal{E}\}$, with $\mathcal{X} = \mathbb{R}^k$. Let $M_{\mathcal{E}}$ satisfy Assumptions 1-3. Let $S_R \subseteq \{1, \dots, k\}$ be the set of variables such that the reward $R(x, a)$ is a function only of $[x]_{S_R}$ (x restricted to the indices in S_R). Then let $S = \mathbf{AN}(R)$ denote the ancestors of S_R in the (fully observable) causal graph corresponding to the transition dynamics of $M_{\mathcal{E}}$. Then the state abstraction $\phi_S(x) = [x]_S$ is a model-irrelevance abstraction for every $e \in \mathcal{E}$.

Proof. To prove that ϕ_S is a model-irrelevance abstraction, we must first show that $r(x) = r(x')$ for any $x, x' : \phi_S(x) = \phi_S(x')$. For this, we note that $\mathbb{E}[R(x)] = \int_{r \in \mathbb{R}} r dp(r|x) = \int_{r \in \mathbb{R}} r dp(r|[x]_S, [x]_{S^c})$ and, because by definition $S^c \subset \mathbf{PA}(R)^c$, we have that $R \perp [x]_{S^c}$. Therefore,

$$\mathbb{E}[R(x)] = \int_{r \in \mathbb{R}} r dp(r|[x]_S) = \int_{r \in \mathbb{R}} r dp(r|[x']_S) = \mathbb{E}[R(x')]. \quad (7)$$

To show that $[x]_S$ is a MISA, we must also show that for any x_1, x_2 such that $\phi(x_1) = \phi(x_2)$, and for any $e \in \mathcal{E}$, the distribution over next state equivalence classes will be equal for x_1 and x_2 .

$$\sum_{x' \in \phi^{-1}(\bar{X})} P_{x_1 x'}^e = \sum_{x' \in \phi^{-1}(\bar{X})} P_{x_2 x'}^e.$$

For this, it suffices to observe that S is closed under taking parents in the causal graph, and that by construction environments only contain interventions on variables outside of the causal set. Specifically, we observe that the probability of seeing any particular equivalence class $[x']_S$ after state x is only a function of $[x]_S$.

$$P([x']_S | x) = f([x]_S, [x']_S)$$

This allows us to define a natural decomposition of the transition function as follows.

$$P(x'|x) = P\left([x]_S \oplus [x]_{S^c} \left| [x']_S \oplus [x']_{S^c} \right.\right) \text{ which by the independent noise assumption gives}$$

$$P(x'|x) = f([x']_S, [x]_S) P([x']_{S^c} | x)$$

We further observe that since the components of x are independent, $\sum_{[x']_{S^c}} P([x']_{S^c} | x) = 1$. We now return to the

property we want to show:

$$\begin{aligned}
 \sum_{x' \in \phi^{-1}(\bar{x})} P_{x_1 x'}^e &= \sum_{x' \in \phi^{-1}(\bar{x})} f([x_1]_S, [x']_S) P(x' | x_1) \\
 &= f(\phi(x_1), \bar{x}) \sum_{[x']_{S^c}} P\left([x']_{S^c} \mid x_1\right) \\
 &= f(\phi(x_1), \bar{x})
 \end{aligned}$$

and because $\phi(x_1) = \phi(x_2)$, we have

$$= f(\phi(x_2), \bar{x})$$

for which we can apply the previous chain of equalities backward to obtain

$$= \sum_{x' \in \phi^{-1}(\bar{x})} P_{x_2 x'}^e$$

□

Proposition 1 (Identifiability and Uniqueness of Causal State Abstraction). *In the setting of the previous theorem, assume the transition dynamics and reward are linear functions of the current state. If the training environment set $\mathcal{E}_{\text{train}}$ satisfies any of the conditions of Theorem 2 (Peters et al., 2016) with respect to each variable in $\mathbf{AN}(R)$, then the causal feature set ϕ_S is identifiable. Conversely, if the training environments don't contain sufficient interventions, then it may be that there exists a ϕ such that ϕ is a model irrelevance abstraction over $\mathcal{E}_{\text{train}}$, but not over \mathcal{E} globally.*

Proof. The proof of the first statement follows immediately from the iterative application of the identifiability result of Peters et al. (2016) to each variable in the causal variables set.

For the converse, we consider a simple counterexample in which one variable x_m is constant in every training environment, with value v_m . Then letting $S = \mathbf{AN}(R)$, we observe that $S \cup \{m\}$ is also a model-irrelevance state abstraction.

First, we show $r(x_1) = r(x_2)$ for any $x_1, x_2 : \phi_{S \cup \{m\}}(x_1) = \phi_{S \cup \{m\}}(x_2)$.

$$\begin{aligned}
 p(R | x_1, a) &= p(R | x_1 | S, a) \\
 &= p(R | x_1 |_{S \cup \{m\}}, a, m = v_m) \\
 &= p(R | (x_2 |_{S \cup \{m\}}, a, m = v_m) \\
 &= p(R | x_2, a)
 \end{aligned}$$

Finally, we must show that

$$\sum_{x' \in \phi_{S \cup \{m\}}^{-1}(\bar{x})} P_{x_1 x'} = \sum_{x' \in \phi_{S \cup \{m\}}^{-1}(\bar{x})} P_{x_2 x'}.$$

Again starting from the result of Theorem 1 we have:

$$\begin{aligned}
 \sum_{x' \in \phi_{S \cup \{m\}}^{-1}(\bar{x})} P_{x_1 x'} &= \sum_{x' \in \phi_{S \cup \{m\}}^{-1}(\bar{x})} f(x_1 |_{S \cup \{m\}}, x' |_{S \cup \{m\}}) p(x' | x_1 |_{(S \cup \{m\})^c}, m = v_m) \\
 &= f(\phi_{S \cup \{m\}}(x_1), \bar{x}) \sum_{x' \in \phi_{S \cup \{m\}}^{-1}(\bar{x})} p(x' | x_1, m = v_m) \\
 &= f(\phi_{S \cup \{m\}}(x_1), \bar{x})
 \end{aligned}$$

and because $\phi_{S \cup \{m\}}(x_1) = \phi_{S \cup \{m\}}(x_2)$, we have

$$= f(\phi_{S \cup \{m\}}(x_2), \bar{x})$$

for which we can apply the previous chain of equalities backward to obtain

$$= \sum_{x' \in \phi_{S \cup \{m\}}^{-1}(\bar{x})} P_{x_2 x'}$$

However, if one of the test environments contains the intervention $x_m \leftarrow v_m + \mathcal{N}(0, \sigma^2)$, then the distribution over next-states in the new environment will violate the conditions for a model-irrelevance abstraction. \square

Theorem 2. Consider an MDP M , with M' denoting a coarser bisimulation of M . Let ϕ denote the mapping from states of M to states of M' . Suppose that the dynamics of M are L -Lipschitz w.r.t. $\phi(X)$ and that T is some approximate transition model satisfying $\max_s \mathbb{E} \|T(\phi(s)) - \phi(T_M(s))\| < \delta$, for some $\delta > 0$. Let $W_1(\pi_1, \pi_2)$ denote the 1-Wasserstein distance. Then

$$\mathbb{E}_{x \sim M'} [\|T(\phi(x)) - \phi(T_{M'}(x))\|] \leq \delta + 2LW_1(\pi_{\phi(M)}, \pi_{\phi(M')}). \quad (8)$$

We will use the shorthand π for $\pi_{\phi(M)}$, the distribution of state embeddings $\phi(M)$ corresponding to the behaviour policy, and π' for $\pi_{\phi(M')}$ for the distribution of state embeddings $\phi(M')$ given by the behaviour policy.

Proof.

$$\begin{aligned} \mathbb{E}_{x \sim M'} [\|T(\phi(x)) - \phi(T_{M'}(x))\|] &= \mathbb{E}_{x \sim M'} [\min_{y \in X_M} \|T(\phi(x)) - T(\phi(y)) + T(\phi(y)) - \phi(T_M(x))\|] \\ &\leq \mathbb{E}_{x \sim M'} [\min_{y \in X_M} \|T(\phi(x)) - T(\phi(y))\| \\ &\quad + \|T(\phi(y)) - \phi(T_M(y))\| + \|\phi(T_M(y)) - \phi(T_M(x))\|] \end{aligned}$$

Let γ be a coupling over the distributions of $\phi(M')$ and $\phi(M)$ such that $\mathbb{E}_{\gamma(\phi(x), \phi(y))} \|\phi(x) - \phi(y)\| = W_1(\pi, \pi')$

$$\begin{aligned} &\leq \mathbb{E}_{x \sim M'} [\mathbb{E}_{\gamma(\phi(y)|\phi(x))} \|T(\phi(x)) - T(\phi(y))\|] + \delta + L\|x - y\| \\ &\leq \mathbb{E}_{x \sim M'} [\mathbb{E}_{\gamma(\phi(y)|\phi(x))} L\|\phi(x) - \phi(y)\| + \delta + L\|\phi(x) - \phi(y)\|] \\ &= \mathbb{E}_{\gamma(\phi(x), \phi(y))} [L\|\phi(x) - \phi(y)\| + \delta + L\|\phi(x) - \phi(y)\|] \\ &= 2LW_1(\pi, \pi') + \delta \end{aligned}$$

\square

Theorem 4 (Existence of model-irrelevance state abstractions). Let \mathcal{E} denote some family of bisimilar MDPs with joint state space $\mathcal{X}_{\mathcal{E}} = \cup_{e \in \mathcal{E}} X_e$. Let the mapping from states in M_e to the underlying abstract MDP \bar{M} be denoted by f_e . Then if the states in $X_{\mathcal{E}}$ satisfy $x \in X_{e'} \cap X_e \implies f_{e'}(x) = f_e(x)$, then $\phi = \cup f_e$ is a model-irrelevance state abstraction for \mathcal{E} .

Proof. First, note that $\cup f_e$ is well-defined (because each f agrees with the rest on the value of all states appearing in multiple tasks). Then ϕ will be a model-irrelevance abstraction for every MDP M_e because it agrees with f_e (a model-irrelevance abstraction). \square

Theorem 3. Let M be our block MDP and \bar{M} the learned invariant MDP with a mapping $\phi : \mathcal{X} \mapsto \mathcal{Z}$. For any L -Lipschitz valued policy π the value difference is bounded by

$$|Q^{\pi}(x, a) - \bar{Q}^{\pi}(\phi(x), a)| \leq \frac{J_R^{\infty} + \gamma L J_D^{\infty}}{1 - \gamma}. \quad (9)$$

Proof.

$$\begin{aligned}
 & \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} |Q^\pi(x_t, a_t) - \bar{Q}^\pi(\phi(x_t), a_t)| \\
 & \leq \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} |R(\phi(x_t), a, \phi(x_{t+1})) - r(x, a)| + \gamma \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} |\mathbb{E}_{x_{t+1} \sim P(\cdot | x_t, a_t)} V^\pi(x_{t+1}) - \mathbb{E}_{z_{t+1} \sim f(\cdot | \phi(x_t), a_t)} \bar{V}^\pi(z_{t+1})| \\
 & = J_R^\infty + \gamma \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} \left| \mathbb{E}_{x_{t+1} \sim P(\cdot | x_t, a_t)} [V^\pi(x_{t+1}) - \bar{V}^\pi(\phi(x_{t+1}))] + \mathbb{E}_{\substack{x_{t+1} \sim P(\cdot | x_t, a_t) \\ z_{t+1} \sim f(\cdot | \phi(x_t), a_t)}} [\bar{V}^\pi(\phi(x_{t+1})) - \bar{V}^\pi(z_{t+1})] \right| \\
 & \leq J_R^\infty + \gamma \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} \left| \mathbb{E}_{x_{t+1} \sim P(\cdot | x_t, a_t)} [V^\pi(x_{t+1}) - \bar{V}^\pi(\phi(x_{t+1}))] \right| \\
 & \quad + \gamma \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} \left| \mathbb{E}_{\substack{x_{t+1} \sim P(\cdot | x_t, a_t) \\ z_{t+1} \sim f(\cdot | \phi(x_t), a_t)}} [\bar{V}^\pi(\phi(x_{t+1})) - \bar{V}^\pi(z_{t+1})] \right| \\
 & \leq J_R^\infty + \gamma \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} \left| \mathbb{E}_{x_{t+1} \sim P(\cdot | x_t, a_t)} [V^\pi(x_{t+1}) - \bar{V}^\pi(\phi(x_{t+1}))] \right| + \gamma L \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} W(\phi(P(\cdot | x_t, a_t)), f(\cdot | \phi(x_t), a_t)) \\
 & = J_R^\infty + \gamma \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} \left| \mathbb{E}_{x_{t+1} \sim P(\cdot | x_t, a_t)} [V^\pi(x_{t+1}) - \bar{V}^\pi(\phi(x_{t+1}))] \right| + \gamma L J_D^\infty \\
 & \leq J_R^\infty + \gamma \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} \mathbb{E}_{x_{t+1} \sim P(\cdot | x_t, a_t)} |[V^\pi(x_{t+1}) - \bar{V}^\pi(\phi(x_{t+1}))]| + \gamma L J_D^\infty \\
 & \leq J_R^\infty + \gamma \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} |[V^\pi(x_t) - \bar{V}^\pi(\phi(x_t))]| + \gamma L J_D^\infty \\
 & \leq J_R^\infty + \gamma \sup_{x_t \in \mathcal{X}, a_t \in \mathcal{A}} |[Q^\pi(x_{t-1}, a_{t-1}) - \bar{Q}^\pi(\phi(x_{t-1}), a_{t-1})]| + \gamma L J_D^\infty \\
 & = \frac{J_R^\infty + \gamma L J_D^\infty}{1 - \gamma}
 \end{aligned}$$

□

Proposition 2 (Lower bound on abstraction error). *Let f_e be a mapping from $\mathcal{S} \rightarrow \mathcal{X}$. Fix some arbitrary policy ρ and let $v(s)$ denote the value of state s under ρ , with π its stationary distribution. If $\exists e, e', s, s'$ such that $f_e(s) = f_{e'}(s')$ (i.e. different states induce the same observation), then the following bound is a lower bound on the error obtained by a joint state abstraction over all environments.*

$$\min_{\hat{v}} \frac{1}{|\mathcal{E}|} \sum_{e \in \mathcal{E}} \text{err}(\phi(X_e), \hat{v}) \geq \min_{s, s': v(s) \neq v(s')} \left(|v(s) - v(s')| \right) P_{\mathcal{E}} \left((\phi(x) \neq f_e^{-1}(x)) \right) \geq \delta \frac{H(V(S)|X) - 1}{\log |V(S)|} \quad (10)$$

Where

$$\text{err}(\phi(X_e), \hat{v}) := \mathbb{E}_{\pi(X_e)} |\hat{v}(\phi(x)) - v(f_e^{-1}(x))|$$

and

$$\delta = \min_{s, s': v(s) \neq v(s')} \left(|v(s) - v(s')| \right)$$

Proof. (Sketch) The error obtained by state abstraction will be at least the decoding error of values from abstract states scaled by δ . This in turn depends on how effectively it is possible to decode a potentially lossy mapping from observations back to states. This leads to the second inequality, due to Fano, where the entropy $H(V(S)|X)$ is given by marginalization with respect to $v(s)$ of the following probability distributions.

$$\begin{aligned}
 p(x) &= \frac{1}{|\mathcal{E}|} \sum_{s, e} \mathbb{1}[f_e(s) = x] \pi(s) \\
 p(s|x) &= \frac{1}{p(x)} \frac{1}{|\mathcal{E}|} \sum_e \pi(s)
 \end{aligned}$$

□

C. Implementation Details

C.1. Model Learning: Rich Observations

For the model learning experiments we use an almost identical encoder architecture as in Tassa et al. (2018), with two more convolutional layers to the convnet trunk. Secondly, we use ReLU activations after each convolutional layer, instead of ELU. We use kernels of size 3×3 with 32 channels for all the convolutional layers and set stride to 1 everywhere, except of the first convolutional layer, which has stride 2. We then take the output of the convolutional net and feed it into a single fully-connected layer normalized by LayerNorm (Ba et al., 2016). Finally, we add tanh nonlinearity to the 50 dimensional output of the fully-connected layer.

The decoder consists of one fully-connected layer that is then followed by four deconvolutional layers. We use ReLU activations after each layer, except the final deconvolutional layer that produces pixels representation. Each deconvolutional layer has kernels of size 3×3 with 32 channels and stride 1, except of the last layer, where stride is 2.

The dynamics and reward models are all MLPs with two hidden layers with 200 neurons each and ReLU activations.

C.2. Reinforcement Learning

For the reinforcement learning experiments we modify the Soft Actor-Critic PyTorch implementation by Yarats and Kostrikov (2020) and augment with a shared encoder between the actor and critic, the general model f_s and task-specific models f_η^e . The forward models are multi-layer perceptions with ReLU non-linearities and two hidden layers of 200 neurons each. The encoder is a linear layer that maps to a 50-dim hidden representation. We also use L1 regularization on the S latent representation. We add two additional dimensions to the state space, a spurious correlation dimension that is a multiplicative factor of the last dimension of the ground truth state, as well as an environment id. We add Gaussian noise $\mathcal{N}(0, 0.01)$ to the original state dimension, similar to how Arjovsky et al. (2019) incorporate noise in the label to make the task harder for the baseline.

Soft Actor Critic (SAC) (Haarnoja et al., 2018) is an off-policy actor-critic method that uses the maximum entropy framework to derive soft policy iteration. At each iteration, SAC performs soft policy evaluation and improvement steps. The policy evaluation step fits a parametric soft Q-function $Q(x_t, a_t)$ using transitions sampled from the replay buffer \mathcal{D} by minimizing the soft Bellman residual,

$$J(Q) = \mathbb{E}_{(x_t, x_t, r_t, x_{t+1}) \sim \mathcal{D}} \left[\left(Q(x_t, a_t) - r_t - \gamma \bar{V}(x_{t+1}) \right)^2 \right].$$

The target value function \bar{V} is approximated via a Monte-Carlo estimate of the following expectation,

$$\bar{V}(x_{t+1}) = \mathbb{E}_{a_{t+1} \sim \pi} [\bar{Q}(x_{t+1}, a_{t+1}) - \alpha \log \pi(a_{t+1} | x_{t+1})],$$

where \bar{Q} is the target soft Q-function parameterized by a weight vector obtained from an exponentially moving average of the Q-function weights to stabilize training. The policy improvement step then attempts to project a parametric policy $\pi(a_t | x_t)$ by minimizing KL divergence between the policy and a Boltzmann distribution induced by the Q-function, producing the following objective,

$$J(\pi) = \mathbb{E}_{x_t \sim \mathcal{D}} \left[\mathbb{E}_{a_t \sim \pi} [\alpha \log(\pi(a_t | x_t)) - Q(x_t, a_t)] \right].$$

We provide the hyperparameters used for the RL experiments in Table 1.

C.3. IRM Hyperparameter Sweep

We found IRM to be very brittle, even on the original colored MNIST task they presented. We implement the same penalty and learning rate schedule proposed in their paper (Arjovsky et al., 2019), but found that we required a much smaller penalty weight to work. In Figure 8 we show the hyperparameter sweep we performed on `cartpole.swingup` to find one where it started to learn. Note that in the colored MNIST task, they used a penalty weight of 1000, whereas we found no learning to occur until we reduced the penalty weight to 0.01.

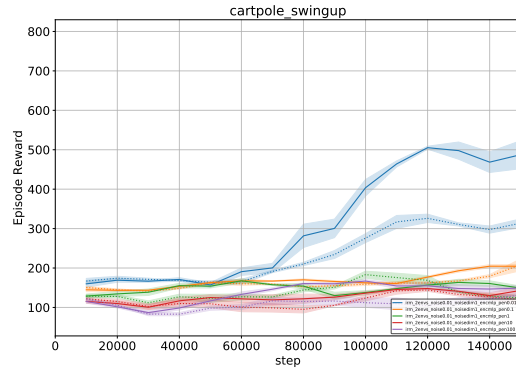


Figure 8. Hyperparameter sweep for IRM on `cartpole_swingup`. All penalty weights fail to learn, even on the training environments, until the penalty weight is very small.

D. Additional Results

D.1. Reinforcement Learning

We find that even without noise on the ground truth states, with only two environments, baseline SAC fails.

Parameter name	Value
Replay buffer capacity	1000000
Batch size	1024
Discount γ	0.99
Optimizer	Adam
Critic learning rate	10^{-5}
Critic target update frequency	2
Critic Q-function soft-update rate τ_Q	0.005
Critic encoder soft-update rate τ_{enc}	0.005
Actor learning rate	10^{-5}
Actor update frequency	2
Actor log stddev bounds	$[-5, 2]$
Encoder learning rate	10^{-5}
Decoder learning rate	10^{-5}
Decoder weight decay	10^{-7}
L1 regularization weight	10^{-5}
Temperature learning rate	10^{-4}
Temperature Adam's β_1	0.9
Init temperature	0.1

Table 1. A complete overview of used hyper parameters.

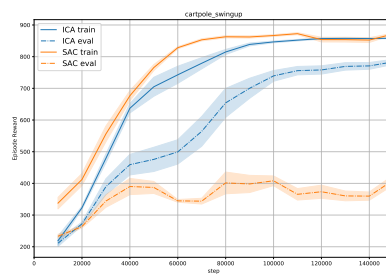


Figure 9. Generalization gap in SAC performance with 2 training environments on Cartpole Swingup from DMC. Evaluated with 10 seeds, standard error shaded.