Curiosity in Hindsight: Intrinsic Exploration in Stochastic Environments

Daniel Jarrett [1] **Corentin Tallec** [1] **Florent Altché** [1] **Thomas Mesnard** [1] **Rémi Munos** [1] **Michal Valko** [1]

Abstract

Consider the problem of exploration in sparse-reward or reward-free environments, such as in Montezuma's Revenge. In the *curiosity-driven* paradigm, the agent is rewarded for how much each realized outcome differs from their predicted outcome. But using predictive error as intrinsic motivation is fragile in *stochastic environments*, as the agent may become trapped by high-entropy areas of the state-action space, such as a "noisy TV". In this work, we study a natural solution derived from structural causal models of the world: Our key idea is to learn representations of the future that capture precisely the *unpredictable* aspects of each outcome—which we use as additional input for predictions, such that intrinsic rewards only reflect the *predictable* aspects of world dynamics. First, we propose incorporating such hindsight representations into models to disentangle "noise" from "novelty", yielding *Curiosity in Hindsight*: a simple and scalable generalization of curiosity that is robust to stochasticity. Second, we instantiate this framework for the recently introduced `BYOL-Explore` algorithm as our prime example, resulting in the noise-robust `BYOL-Hindsight`. Third, we illustrate its behavior under a variety of different stochasticities in a grid world, and find improvements over `BYOL-Explore` in hard-exploration Atari games with sticky actions. Notably, we show state-of-the-art results in exploring Montezuma's Revenge with sticky actions, while preserving performance in the non-sticky setting.

1. Introduction

Learning to understand the world without supervision is a hallmark of intelligent behavior [1], and *exploration* is a key pillar of research in reinforcement learning agents [2]. How might an agent learn meaningful behaviors when external rewards are sparse or absent? A predominant approach is given by the *curiosity-driven* paradigm [3], in which an agent's ability to predict the future is used as a proxy for their "understanding" of the world. Maintaining a learned model of the environment, at each step the agent receives an intrinsic reward proportional to how much the realized outcome differs from their predicted outcome—which naturally directs them towards new areas that have not been seen.

There are two major hurdles. The first concerns *dimensionality*: While outcomes can be predicted directly at the level of observations [4–8], pixel-based losses have generally not worked well in higher dimensions [9]. Popular solutions thus operate on lower-dimensional *latent representations*, such as frame-predictive features [10], inverse dynamics features [11], random features [12], or features that maximize information across time [13]. Most recently, bootstrapped features are employed in `BYOL-Explore` [14]—achieving superhuman performance on hard-exploration games in Atari with a much simpler design than comparable agents.

The second concerns *stochasticity*, which is our focus here: Curiosity-driven agents are often susceptible to bad behavior in environments with stochastic transitions, since they are often hopelessly attracted to high-entropy elements in the state-action space [9]. A classic example is the problem of a "noisy TV", which generates a stream of intrinsic rewards around which predictive error-based agents become stuck indefinitely [15]. More generally, this problem manifests with respect to any aspect of environment dynamics that is inherently unpredictable, including noise specific to certain states, as well as noise actively induced by the agent.

Novelty vs. Noise In the presence of stochasticity, predictive error *per se* is no longer a good measure for an agent's lack of "understanding" of the world. Intuitively, we wish to measure their understanding by how much *epistemic* knowledge they have acquired (viz. necessary truths about how the world works in general), which is entirely orthogonal to how much *aleatoric* variation each outcome can display (viz. contingent facts about how the world happens to be). Precisely, we want to distinguish between aspects of world dynamics that are inherently predictable—for which (reducible) errors stem from "novelty"—and aspects that are inherently unpredictable—for which (irreducible) errors stem from "noise". Crucially, while the former should contribute to intrinsic rewards for exploration, the latter should not.

[1]DeepMind. Correspondence: Dan Jarrett <jarrettd@google.com>.

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

Contributions We operationalize this distinction by deriving a solution based on structural causal models of the world: Our key idea is to learn representations of the future that capture precisely the unpredictable aspects of each outcome—no more, no less—which we use as additional input for predictions, such that intrinsic rewards vanish in the limit. First, we propose incorporating such hindsight representations into the agent's model to disentangle "noise" from "novelty", yielding *Curiosity in Hindsight*: a simple and scalable generalization of curiosity-driven exploration that is robust to stochasticity (Section 3). Second, we instantiate this framework for the recently introduced `BYOL-Explore` algorithm as our prime example, giving rise to the noise-robust `BYOL-Hindsight` (Section 4). Third, we illustrate its behavior under a variety of different stochasticities in a grid world, and find improvements over `BYOL-Explore` in hard-exploration Atari games with sticky actions (a standard protocol for introducing stochasticity in training/evaluation). Notably, we show state-of-the-art results in exploring Montezuma's Revenge with sticky actions, while preserving its original performance in the non-sticky setting (Section 5).

2. Motivation

2.1. Problem Formalism

Consider the standard MDP setup. We employ uppercase for random variables and lowercase for specific values: Let X denote the *state* variable, taking on values $x \in \mathcal{X}$, and A the *action* variable, taking on values $a \in \mathcal{A}$. While we keep notation simple, X may play the role of "contexts", "features", "embeddings", or "beliefs" depending on environment observability and the design of the agent. Let $\tau \in \Delta(\mathcal{X})^{\mathcal{X}\times\mathcal{A}}$ denote the world's dynamics such that $X_{t+1} \sim \tau(\cdot|x_t, a_t)$, and $\pi \in \Delta(\mathcal{A})^{\mathcal{X}}$ the agent's policy such that $A_t \sim \pi(\cdot|x_t)$. Lastly, let ρ_π denote the distribution of states induced by π.

Definition 1 (Curiosity-driven Exploration) In this work we focus on *predictive error-based* curiosity—subsuming most popular approaches to curiosity. Intrinsic rewards are:

$$\mathcal{R}_\eta(x_t, a_t) \coloneqq -\mathbb{E}_{X_{t+1}\sim\tau(\cdot|x_t,a_t)} \log \tau_\eta(X_{t+1}|x_t, a_t) \quad (1)$$

where τ_η is the agent's world model[2] parameterized by η, and is trained using the trajectories collected by rolling out a policy that seeks to maximize this same prediction error:

$$\underset{\pi}{\overset{\text{(policy)}}{\text{maximize}}} \ \underset{\eta}{\overset{\text{(model)}}{\min}} \ \mathbb{E}_{\substack{X_t\sim\rho_\pi \\ A_t\sim\pi(\cdot|X_t)}} \mathcal{R}_\eta(X_t, A_t) \quad (2)$$

Example 1 (Bootstrapping Representations) As our key example, recall `BYOL-Explore` [14], a most recent and successful incarnation of this paradigm. The *prediction loss* for a given transition (x_t, a_t, x_{t+1}) is defined as the following:

$$\mathcal{L}_\eta^{\texttt{BYOL}}(x_t, a_t, x_{t+1}) \coloneqq \left\| x_{t+1} - \hat{x}_{t+1} \right\|_2^2 \quad (3)$$

and the (state-action) *prediction bonus* for the agent's policy:

$$\mathcal{R}_\eta^{\texttt{BYOL}}(x_t, a_t) \coloneqq \mathbb{E}_{X_{t+1}\sim\tau(\cdot|x_t,a_t)} \mathcal{L}_\eta^{\texttt{BYOL}}(x_t, a_t, X_{t+1}) \quad (4)$$

where the novelty of the method lies in the manner in which specific quantities are defined and learned: **(i.)** Input states are RNN "belief" representations $x_t \coloneqq b_t$ of previous actions $\{a_{t'}\}_{t'<t}$ and observation encodings $\{\omega(o_{t'})\}_{t'\leq t}$, where ω is an encoding function; **(ii.)** Target states are ℓ_2-normalized encodings $x_{t+1} \coloneqq \texttt{sg}(\omega_{\text{target}}(o_{t+1})/\|\omega_{\text{target}}(o_{t+1})\|_2)$ of future observations, with ω_{target} being an exponential moving average of ω, and `sg` denotes the stop-gradient operator; and **(iii.)** Predictions are ℓ_2-normalized transformations of current beliefs and actions: $\hat{x}_{t+1} \coloneqq h_\eta(b_t, a_t)/\|h_\eta(b_t, a_t)\|_2$, where h_η is a prediction function. Multi-step open-loop predictions are a straightforward extension. See Appendix C for a more detailed review of the `BYOL-Explore` algorithm.

Stochastic Traps In stochastic environments, Equation 1 does not converge to zero even with infinite experience: It converges to the entropy $\mathbb{H}[X_{t+1}|x_t, a_t]$, so the agent may become stuck on repeatedly experiencing (intrinsically rewarding) transitions where entropy is high. Instead, what we desire is a reward that converges to zero in the limit. The notion of "optimistic" exploration offers a hint of what might be possible—Consider constructing a reward that satisfies:

$$\mathcal{R}_\eta(x_t, a_t) \geq D_{\text{KL}}\big(\tau(X_{t+1}|x_t, a_t) \| \tau_\eta(X_{t+1}|x_t, a_t)\big) \quad (5)$$

which upper bounds the distance between the world and the agent's model. On the one hand, while Definition 1 verifies this, the bound fails to tighten even in the limit. On the other hand, it is hard to measure this distance directly, since the entropy term is by construction unknown. As it turns out, we shall later see that our proposed technique effectively gives a reward that verifies the inequality—and is tight in the limit.

2.2. Related Work

Our work inherits from the curiosity-driven paradigm [3–18], among which some methods have been designed with robustness to certain stochasticities in mind (Table 1). However, our method is uniquely characterized by the following:

1. **Stochasticity Types**: First, it is capable of handling all types of stochasticities in generality. Specifically, this includes stochasticity that is *entirely random* (e.g. a viewport polluted by noise sampled according to a distribution independent of states and actions), stochasticity that is *state-dependent* (e.g. a visible object that performs a random walk within the environment), as well as *action-dependent* (e.g. a layer of random pixels that only appears if sampled on demand by specific actions). For instance, previous works have found that inverse dynamics features can learn to filter out random noise [11], but may break down in the presence of action-dependent noise [9, 19].

[2] Note that this is only used for computing rewards, and need not be related to the underlying RL algorithm, which can be model-free.

Table 1. Relationship with Curiosity-driven Exploration. Curiosity in Hindsight is a drop-in modification on top of any prediction error-based method, and is characterized by being robust to different noises, being dynamics aware, and being general to any representation space.

Curiosity-driven Exploration Method	*Prediction Inputs*	*Prediction Target*	*Measure of Learning*	Random Noise	X-/A-Dep. Noise	Dynamics Awareness	**Representation Space**
AE [10]	X_t, A_t	X_{t+1}	$\mathcal{L}_\eta^{\text{predict}}$	✗	✗	✓	reconstructive
ICM [11]	X_t, A_t	X_{t+1}	$\mathcal{L}_\eta^{\text{predict}}$	✓	✗	✓	action predictive
EMI [13]	X_t, A_t	X_{t+1}	$\mathcal{L}_\eta^{\text{predict}}$	✗	✗	✓	MI-maximizing
RND [12]	X_{t+1}	$f_{\text{random}}(X_{t+1})$	$\mathcal{L}_\eta^{\text{predict}}$	✓	✓	✗	random projection
Dora [16]	X_t, A_t	const. zero	$\mathcal{L}_\eta^{\text{predict}}$	✓	✓	✗	pixel space
AMA [15]	X_t, A_t	X_{t+1}	$\mathcal{L}_\eta^{\text{predict}} - \text{Tr}(\hat{\Sigma}_{t+1})$	✓	✓	✓	pixel space
`BYOL-Explore` [14]	X_t, A_t	X_{t+1}	$\mathcal{L}_\eta^{\text{predict}}$	✗	✗	✓	bootstrapped
Curiosity in Hindsight (e.g. `BYOL-Hindsight`)	X_t, A_t, Z_{t+1}	X_{t+1}	$\mathcal{L}_{\theta,\eta}^{\text{reconstruct}} + \mathcal{L}_{\theta,\nu}^{\text{invariance}}$	✓	✓	✓	***any representation***

2. **Dynamics Awareness**: Second, it does not require entirely discarding dynamics learning. By way of contrast, consider purely frequency-oriented exploration strategies, such as learning to predict a random projection of observations [12], or simply to predict the constant zero [16]. As these are deterministic functions of their inputs, they are in principle resilient to stochasticity. But empirically they can still behave poorly in the presence of action-dependent stochasticities [15]: If the noise is sufficiently *diffuse*, the agent may never learn the function well, so in the absence of any other learning signal—such as the dynamics of the world—they may still become stuck [20].

3. **Generality and Scalability**: As a drop-in modification, it is generally be applicable to any underlying choice of representation space. In contrast, existing techniques capable of handling stochasticity are often tied to specific feature spaces, such as to employ inverse dynamics features [11], random features [12], or pixel-space features [15]—which may limit their flexibility of application. Moreover, unlike ensemble-based or disagreement-based techniques that require training a large number of models [19, 21, 22],[3] we shall see that incorporating hindsight is simpler and more *scalable* by only requiring the addition of an auxiliary component to the usual prediction loss.

Alternative paradigms for exploration have been proposed: Novelty-based methods encourage exploration on the basis of visitation counts [25], hashes [26], density estimates [27–30], and adversarial guidance [31, 32]; further extensions account for episodic memory [33–35] and the long-term value of exploratory actions [16, 36–38]. Knowledge-based methods encourage exploration on the basis of the agent's uncertainty about the world [19, 39], with most work focusing on estimating the information gain from different actions [21,22,40–46], or directly estimating learning progress [47–49]. Finally, diversity-based methods seek to maximize the state entropy [50–53], or to encourage learning diverse skills [54–65] and reaching different goals [66–70].

[3]Ensembles can in principle approximate uncertainty [19,21–23]; in practice, training and scaling is difficult with larger architectures, and models often converge prematurely to the same outputs [24].

3. Curiosity in Hindsight

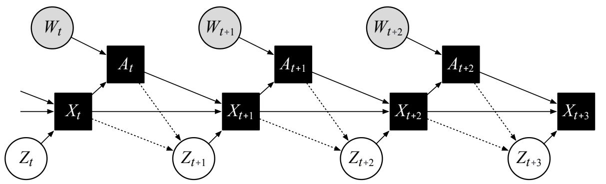
Consider the game of betting on a hidden dice roll: Suppose we take the action $A_t =$ "bet on 6", then observe the outcome $X_{t+1} =$ "lost the bet". Two facts are clear: (1) *a priori*, we could not have predicted this result at all; (2) *a posteriori*, we may deduce the (latent) fact $Z_{t+1} =$ "the die must have rolled 1–5". These are not contradictory. In particular, the former does *not* imply that we lack an understanding of how the game works, nor does it suggest that we should engage in further such bets to improve our understanding. Indeed, knowing how the game works, in hindsight (i.e. given what we deduced about Z_{t+1}), the outcome is obvious to us (i.e. we can now deterministically identify X_{t+1}). Conversely, suppose we actually *didn't* know how the game works: Then we couldn't have correctly inferred Z_{t+1}, nor would its knowledge have enabled us to identify X_{t+1} with any certainty. If so, engaging in additional bets may indeed allow us to learn and improve our understanding of how they work.

Intuitively, we can thus measure our understanding of each transition based on how much the outcome makes sense *in hindsight*. So, instead of asking "How well can we predict X_{t+1} *a priori*?", we actually want to ask "How well can we reconstruct X_{t+1} *a posteriori*—i.e. given hindsight Z_{t+1}?". First, we formalize this intuition using the language of *posterior inference* when a known model of the world is available (Section 3.1). Then we generalize this approach to generating learned *hindsight representations* when the world model needs to be learned at the same time (Section 3.2). Finally, we derive *Curiosity in Hindsight* on the basis of these ingredients, showing it approximates optimistic exploration (Inequality 5) while being robust to stochasticities (Section 3.3).

3.1. Structural Causal Model

Let Z denote a *latent* variable, taking on values $z \in \mathcal{Z}$. For each observed transition (x_t, a_t, x_{t+1}), we let z_{t+1} encapsulate *all* sources of unobserved stochasticity in the dynamics. By construction, $x_{t+1} = f(x_t, a_t, z_{t+1})$ for some deterministic function f, and a prior p over Z_{t+1} induces the environment dynamics—that is, the distribution $\tau(X_{t+1}|x_t, a_t)$. Figure 1 illustrates the structural causal model, with solid

Figure 1. Structural Causal Model. By the reparameterization lemma, there exists an equivalent graphical representation under which all stochasticities are exogenous (i.e. dotted edges removed).



squares for deterministic nodes, shaded circles for observable stochastic nodes, and unshaded circles for unobservable stochastic nodes (W captures any randomness in the policy). In general, stochasticities can be entirely random (i.e. no edges into Z_{t+1}), state-dependent (i.e. edge $X_t \rightarrow Z_{t+1}$), or action-dependent (i.e. edge $A_t \rightarrow Z_{t+1}$). However, by the *reparameterization lemma* it is always possible to represent an environment such that all stochasticities are effectively exogenous [71–73] (i.e. no directed edges into Z_{t+1}).

From Prediction to Reconstruction Consider the setting in which we know the model f. Suppose first that we somehow had access to each latent z_{t+1}. Then the outcome of a transition at state x_t and action a_t would be deterministically computable with no uncertainty (i.e. reconstruction error = zero):

$$x_{t+1} \equiv f(x_t, a_t, z_{t+1}) \tag{6}$$

In reality, the latent variable z_{t+1} is not observed. Thus it may seem like the best we can do is to compute the *a priori* expectation of the outcome (i.e. prediction error = entropy):

$$\mathbb{E}_{X_{t+1}\sim\tau(\cdot|x_t,a_t)} X_{t+1} \equiv \mathbb{E}_{Z_{t+1}\sim p} f(x_t, a_t, Z_{t+1}) \tag{7}$$

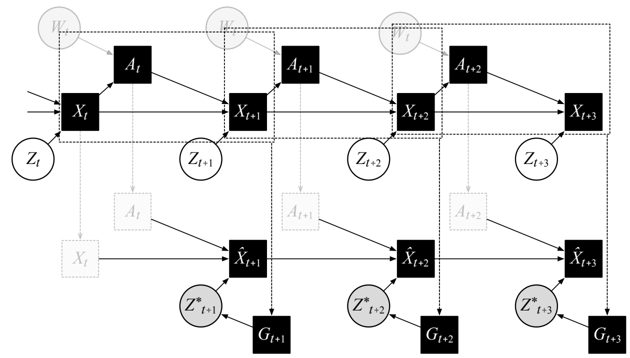
However, while z_{t+1} is not observable, based on the transition (x_t, a_t, x_{t+1}) we can infer *a posteriori* what its values could have been. Importantly, by the consistency property of counterfactuals we know $f(x_t, a_t, Z_{t+1}) = x_{t+1}$ for any $Z_{t+1} \sim p(\cdot|x_t, a_t, x_{t+1})$ [74]. That is to say, conditioned on hindsight, the reconstruction error of the true model is zero. This suggests when f is unknown and learned by the agent, *the reconstruction error may be an attractive candidate for an intrinsic reward*. Of course, now the missing piece is how to sample Z_{t+1} from the posterior—which we discuss next.

3.2. Hindsight Representations

Realistically, the model $f(X_t, A_t, Z_{t+1})$ is unknown, so we learn to approximate it using a *reconstructor* f_η parameterized by η.[4] Exact posterior inference $p_\eta(Z_{t+1}|X_t, A_t, X_{t+1})$ is intractable, so we learn to approximate it using a *generator* p_θ parameterized by θ. Two objectives are key. First, as noted above, representations Z_{t+1} should be *reconstructive* of outcomes X_{t+1}. Here we can simply use the squared loss:

[4] In general the model is not identifiable and there is no guarantee that $f_\eta(x_t, a_t, Z_{t+1})$ be close to x_{t+1} for arbitrary Z_{t+1}. But we are not interested in making counterfactual queries: For reconstruction, we only wish to evaluate f_η where $Z_{t+1} \sim p_\eta(\cdot|x_t, a_t, x_{t+1})$.

Figure 2. Hindsight Representations. A learned generator $G_{t+1} \coloneqq p_\theta(\cdot|x_t, a_t, x_{t+1})$ generates hindsight vectors—denoted Z^*_{t+1} in this figure to be distinguished from "ground-truth" latents Z_{t+1}.



Objective 1 (Reconstruction) Let the *reconstruction loss* for a given transition $(x_t, a_t, z_{t+1}, x_{t+1})$—including hindsight representation z_{t+1} drawn from $p_\theta(\cdot|x_t, a_t, x_{t+1})$—be:

$$\mathcal{L}^{\text{rec.}}_\eta(x_t, a_t, z_{t+1}, x_{t+1}) \coloneqq \left\| x_{t+1} - f_\eta(x_t, a_t, z_{t+1}) \right\|_2^2 \tag{8}$$

and (state-action) *reconstruction bonus* for the agent policy:

$$\mathcal{R}^{\text{rec.}}_{\theta,\eta}(x_t, a_t) \coloneqq \mathbb{E}_{\substack{X_{t+1}\sim\tau(\cdot|x_t,a_t) \\ Z_{t+1}\sim p_\theta(\cdot|x_t,a_t,X_{t+1})}} \mathcal{L}^{\text{rec.}}_\eta(x_t, a_t, Z_{t+1}, X_{t+1}) \tag{9}$$

Driven to zero, this requires hindsight representations to encapsulate *at least* all aspects of the world's dynamics that are unpredictable (so that we *don't* reward the agent for irreducible error). However, we also don't want Z_{t+1} to simply leak information about the outcome that is actually predictable to begin with (so that we *do* reward the agent for reducible error).[5] Thus our second objective requires it to be *independent* of X_t, A_t. Denote the pointwise mutual information between state-action x_t, a_t and hindsight z_t by $\text{PMI}_\theta(x_t, a_t; z_{t+1}) \coloneqq \log(p_\theta(z_{t+1}|x_t, a_t)/p_\theta(z_{t+1}))$. Then:

Objective 2 (Invariance) Let the *invariance loss* for a given transition $(x_t, a_t, z_{t+1}, x_{t+1})$—again, where the hindsight representation z_{t+1} is drawn from $p_\theta(\cdot|x_t, a_t, x_{t+1})$—be:

$$\mathcal{L}^{\text{inv.}}_\theta(x_t, a_t, z_{t+1}) \coloneqq \text{PMI}_\theta(x_t, a_t; z_{t+1}) \tag{10}$$

and (state-action) *invariance bonus* for the agent's policy:

$$\mathcal{R}^{\text{inv.}}_\theta(x_t, a_t) \coloneqq \mathbb{E}_{\substack{X_{t+1}\sim\tau(\cdot|x_t,a_t) \\ Z_{t+1}\sim p_\theta(\cdot|x_t,a_t,X_{t+1})}} \mathcal{L}^{\text{inv.}}_\theta(x_t, a_t, Z_{t+1}) \tag{11}$$

Driven to zero, this requires hindsight representations to encapsulate *at most* all aspects of the world's dynamics that are unpredictable. This suggests a combination—of reconstruction loss (of a dynamics model) plus invariance loss (of a hindsight model)—may make a good signal for exploration. We now have all the ingredients required for Curiosity in Hindsight, which it is instructive to contrast with Definition 1:

[5] For example, consider the solution $Z_{t+1} \coloneqq X_{t+1}$, where reconstruction error is trivially zero, but is pathological for exploration.

3.3. Optimistic Exploration

Definition 2 (Curiosity in Hindsight) Let the *hindsight intrinsic reward function* be defined as the weighted combination of Objectives 1 and 2, with a tradeoff coefficient λ:

$$\mathcal{R}_{\theta,\eta}(x_t, a_t) \coloneqq \frac{1}{\lambda}\mathcal{R}^{\text{rec.}}_{\theta,\eta}(x_t, a_t) + \mathcal{R}^{\text{inv.}}_{\theta}(x_t, a_t) \tag{12}$$

and the dynamics and hindsight models are jointly trained to minimize this quantity over the trajectories it collects, while rolling out a policy seeking to maximize this same quantity:

$$\underset{\pi}{\overset{\text{(policy)}}{\text{maximize}}} \; \underset{\theta,\eta}{\overset{\text{(model)}}{\min}} \; \mathbb{E}_{\substack{X_t\sim\rho_\pi \\ A_t\sim\pi(\cdot|X_t)}} \mathcal{R}_{\theta,\eta}(X_t, A_t) \tag{13}$$

Recall that in the presence of stochastic transitions, standard curiosity-driven exploration can be seen as a poor approximation to "optimistic" exploration (Inequality 5)—because the bound is never tight even in the limit, which renders it susceptible to stochastic traps. The following result shows that exploration by Curiosity in Hindsight can resolve this:

Theorem 1 (Optimistic Exploration) Let the coefficient λ satisfy the inequality $\frac{1}{2}\log(\lambda\pi) \leq \mathbb{H}_\theta[X_{t+1}|x_t, a_t, Z_{t+1}] + D_{\text{KL}}\big(p_\theta(Z_{t+1}|x_t, a_t)\|p_\theta(Z_{t+1})\big)$, where π denotes here the mathematical constant (not the agent's policy). Then:

$$\mathcal{R}_{\theta,\eta}(x_t, a_t) \geq D_{\text{KL}}\big(\tau(X_{t+1}|x_t, a_t)\|\tau_{\theta,\eta}(X_{t+1}|x_t, a_t)\big) \tag{14}$$

where $\tau_{\theta,\eta}(X_{t+1}|x_t, a_t) \coloneqq \mathbb{E}_{Z_{t+1}\sim p_\theta} p_\eta(X_{t+1}|x_t, a_t, Z_{t+1})$ denotes the learned world model. Furthermore, assuming realizability, rewards vanish at optimal parameters θ^*, η^*:

$$\mathcal{R}_{\theta^*,\eta^*}(x_t, a_t) = 0 \quad \forall x_t, a_t \in \text{supp}(\rho_\pi) \tag{15}$$

Proof. Appendix A. □

In other words, by choosing a small enough λ term, the hindsight intrinsic reward (Equation 12) is an upper bound on the KL-term we care about (Inequality 5). Driving the reward to zero also drives the KL to zero, thus the reward-maximizing exploration policy (Equation 13) is approximating precisely the sort of "optimistic" exploration that we desired to start.

4. Practical Framework

Two questions remain. Firstly, how should the invariance terms be computed? For this, we propose a contrastive learning framework to approximate them (Section 4.1). Secondly, what does a concrete implementation look like? For this, we instantiate this framework on top of `BYOL-Explore`, yielding its robust variant `BYOL-Hindsight` (Section 4.2).

4.1. Contrastive Learning

To estimate the pointwise mutual information, we use an auxiliary *critic* g_ν parameterized by ν, trained as maximizer:

Objective 3 (Contrastive Learning) Let the *contrastive loss* for a transition, with respect to a batch of $K-1$ negative hindsight samples $z^{1:K-1}_{t+1} \coloneqq z^1_{t+1}, ..., z^{K-1}_{t+1}$, be defined as:

$$\ell^{K,\text{con.}}_{\theta,\nu}(x_t, a_t, z_{t+1}, z^{1:K-1}_{t+1}) \coloneqq \log \frac{e^{g_\nu(x_t,a_t,z_{t+1})}}{\frac{1}{K}\left(e^{g_\nu(x_t,a_t,z_{t+1})} + \sum_{i=1}^{K-1} e^{g_\nu(x_t,a_t,Z^i_{t+1})}\right)} \tag{16}$$

such that the overall contrastive loss for the transition is its expectation over negative hindsight samples from rollouts:

$$\mathcal{L}^{K,\text{con.}}_{\theta,\nu}(x_t, a_t, z_{t+1}) \coloneqq \mathbb{E}_{\substack{(X^1_t,...,X^{K-1}_t)\sim\prod_{i=1}^{K-1}\rho_\pi \\ (A^1_t,...,A^{K-1}_t)\sim\prod_{i=1}^{K-1}\pi(\cdot|X^i_t) \\ (X^1_{t+1},...,X^{K-1}_{t+1})\sim\prod_{i=1}^{K-1}\tau(\cdot|X^i_t,A^i_t) \\ (Z^1_{t+1},...,Z^{K-1}_{t+1})\sim\prod_{i=1}^{K-1}p_\theta(\cdot|X^i_t,A^i_t,X^i_{t+1})}} \ell^{K,\text{con.}}_{\theta,\nu}(x_t, a_t, z_{t+1}, z^{1:K-1}_{t+1}) \tag{17}$$

and (state-action) *contrastive bonus* for the agent's policy:

$$\mathcal{R}^{K,\text{con.}}_{\theta,\nu}(x_t, a_t) \coloneqq \mathbb{E}_{\substack{X_{t+1}\sim\tau(\cdot|x_t,a_t) \\ Z_{t+1}\sim p_\theta(\cdot|x_t,a_t,X_{t+1})}} \mathcal{L}^{K,\text{con.}}_{\theta,\nu}(x_t, a_t, Z_{t+1}) \tag{18}$$

How does Objective 3 approximate Objective 2? Precisely:

Theorem 2 (Optimal Invariance) The contrastive bonus lower-bounds the (ideal) invariance bonus for any pair x_t, a_t:

$$\mathcal{R}^{K,\text{con.}}_{\theta,\nu}(x_t, a_t) \leq \mathcal{R}^{\text{inv.}}_{\theta}(x_t, a_t) \tag{19}$$

Furthermore, assuming realizability, for optimal critic parameter $\nu^*_K \coloneqq \arg\max_\nu \mathbb{E}_{X_t,A_t\sim\rho_\pi}\mathcal{R}^{K,\text{con.}}_{\theta,\nu}(X_t, A_t)$ the bound is asymptotically tight (in the batch size $K \to \infty$):

$$\lim_{K\to\infty} \mathcal{R}^{K,\text{con.}}_{\theta,\nu^*_K}(x_t, a_t) = \mathcal{R}^{\text{inv.}}_{\theta}(x_t, a_t) \tag{20}$$

Proof. Appendix A. □

Practical Algorithm This suggests a straightforward algorithm. In practice, batch size $K < \infty$ and critic ν is not fully optimized. The intrinsic reward (Definition 2) now becomes:

$$\mathcal{R}^{K}_{\theta,\eta,\nu}(x_t, a_t) \coloneqq \frac{1}{\lambda}\mathcal{R}^{\text{rec.}}_{\theta,\eta}(x_t, a_t) + \mathcal{R}^{K,\text{con.}}_{\theta,\nu}(x_t, a_t) \tag{21}$$

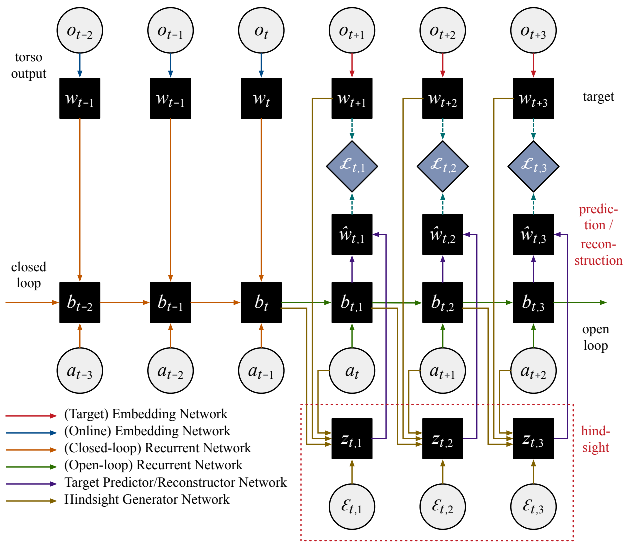
and optimization alternates between training the critic (maximizer) and dynamics and hindsight models (minimizers):

$$\underset{\pi}{\overset{\text{(policy)}}{\text{maximize}}} \; \underset{\theta,\eta}{\overset{\text{(model)}}{\min}} \; \max_{\nu} \; \mathbb{E}_{\substack{X_t\sim\rho_\pi \\ A_t\sim\pi(\cdot|X_t)}} \mathcal{R}^{K}_{\theta,\eta,\nu}(X_t, A_t) \tag{22}$$

Overall, our framework constitutes a simple drop-in modification on top of any curiosity-driven method: Instead of learning a *predictive model* specifying $X_{t+1}\sim\tau_\eta(\cdot|X_t, A_t)$, we now learn a (hindsight-augmented) *reconstructive model* specifying $X_{t+1} = f_\eta(X_t, A_t, Z_{t+1})$. The main ingredients include the reconstructor $f_\eta(X_t, A_t, Z_{t+1})$, the generator $p_\theta(Z_{t+1}|X_t, A_t, X_{t+1})$, and the critic $g_\nu(X_t, A_t, Z_{t+1})$, and the main hyperparameters are the contrastive batch size K and the coefficient λ in the hindsight intrinsic reward.

Finally, note that for ease of exposition we have focused on modeling single-step transitions; however, it is straightforward to generalize this approach to the case of multi-step reconstruction horizons using open-loop rollouts (Figure 2).

Figure 3. From BYOL-Explore to BYOL-Hindsight. The latter simply replaces the prediction loss with our reconstruction and contrastive losses (through the addition of hindsight), with no change to the underlying (bootstrapped) representation learning method.



4.2. BYOL-Hindsight

The preceding discussion used MDP notation, but in practice input states are often representations of histories, and observations are often represented in latent space. Recall `BYOL-Explore` (Example 1) is an incarnation of curiosity (Definition 1) that learns such representations. We now augment it with hindsight, giving rise to the novel `BYOL-Hindsight`:

Example 2 (Bootstrapping with Hindsight) Let the *hindsight loss* for a transition $(x_t, a_t, z_{t+1}, x_{t+1})$ be defined as:

$$\mathcal{L}^{K,\,\texttt{BYOL-Hind.}}_{\theta,\eta,\nu}(x_t, a_t, z_{t+1}, x_{t+1}) \coloneqq \frac{1}{\lambda}\mathcal{L}^{\text{rec.}}_{\eta}(x_t, a_t, z_{t+1}, x_{t+1}) + \mathcal{L}^{K,\text{con.}}_{\theta,\nu}(x_t, a_t, z_{t+1}) \tag{23}$$

and the (state-action) *hindsight bonus* for the agent's policy:

$$\mathcal{R}^{K,\,\texttt{BYOL-Hind.}}_{\theta,\eta,\nu}(x_t, a_t) \coloneqq \mathop{\mathbb{E}}_{\substack{X_{t+1}\sim\tau(\cdot|x_t,a_t)\\ Z_{t+1}\sim p_\theta(\cdot|x_t,a_t,X_{t+1})}} \mathcal{L}^{K,\,\texttt{BYOL-Hind.}}_{\theta,\eta,\nu}(x_t, a_t, Z_{t+1}, X_{t+1}) \tag{24}$$

where the key difference from `BYOL-Explore` lies in swapping out predictions for reconstructions. Specifically, input states x_t and target states x_{t+1} are defined just as before: **(i.)** Input states are RNN "belief" representations; and **(ii.)** Target states are ℓ_2-normalized encodings of future observations. However, instead of (target) predictions, we now have (target) reconstructions: **(iii.)** Reconstructions are ℓ_2-normalized transformations of beliefs, actions, and hindsight: $f_\eta(b_t, a_t, z_{t+1}) \coloneqq h_\eta(b_t, a_t, z_{t+1})/\|h_\eta(b_t, a_t, z_{t+1})\|_2$, where h_η is a reconstruction function, and the contrastive loss encourages hindsight Z_{t+1} to be independent of B_t, A_t.

Figure 3 gives the concrete architecture for `BYOL-Explore`/`BYOL-Hindsight`, with the full multi-step horizon setup: First, an *online* embedding network ω encodes observations o_t into representations $w_t = \omega(o_t)$. A *closed-loop* RNN then computes representations b_t of histories up until each time step t. This is used to initialize an *open-loop* RNN that computes representations $b_{t,i}$ for horizon steps indexed as i. These representations are fed to a predictor/reconstructor network to output $\hat{w}_{t,i}$.[6] Finally, prediction/reconstruction targets are encoded with a *target* embedding network that is an exponential moving average of the online network. The prediction/reconstruction error $\mathcal{R}_{t,i}$ at each open-loop step is computed, and the intrinsic reward associated to each observed transition (o_s, a_s, o_{s+1}) is the sum of errors $\sum_{t+i=s+1}\mathcal{R}_{t,i}$. In `BYOL-Hindsight`, the generator samples $Z_{t,i}$ by taking noise $\varepsilon_{t,i}$ as input, and an additional critic (not pictured) encourages $Z_{t,i}$ to be independent of $B_{t,i-1}$ and A_{t+i-1}. See Algorithms 1–2 in Appendix C for details.

5. Experiments

Three questions deserve empirical study: **(a.) Effectiveness**: In stochastic environments, predictive error-based methods —e.g. `BYOL-Explore`—may fail. Does `BYOL-Hindsight` address the problem? **(b.) Robustness**: Is the method robust to the different types of stochasticities—i.e. independent noise, state-dependent noise, and action-dependent noise? **(c.) Non-specificity**: In environments with no stochasticity, hindsight should confer no benefit. Does `BYOL-Hindsight` manage to preserve the performance of `BYOL-Explore`?

Implementation In all experiments, we start from the same architecture/hyperparameters for `BYOL-Explore` as given in [14], including target network EMA, open-loop horizon, intrinsic reward normalization/prioritization, representation sharing, and VMPO [75] as the underlying RL algorithm. `BYOL-Hindsight` begins from the same setup. The generator, reconstructor, and critic networks are MLPs with three hidden layers of 512; the dimension of the generator noise ϵ and hindsight vector is 256; and $\lambda = 1$. Finally, where shown for reference, RND and ICM are also implemented exactly as described in [14]. See Appendix C for additional detail.

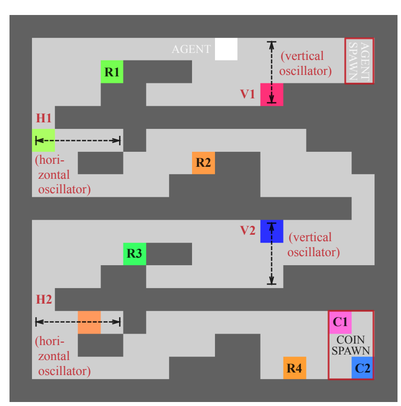


Figure 4. Pycolab Maze Environment Map.

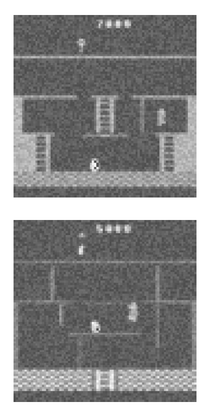
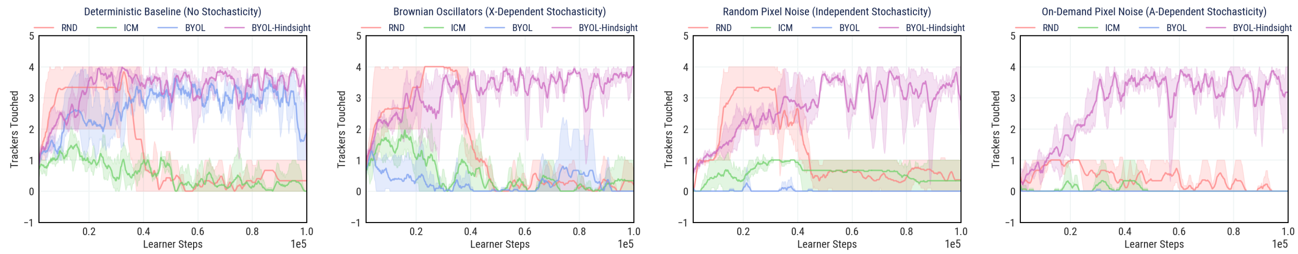


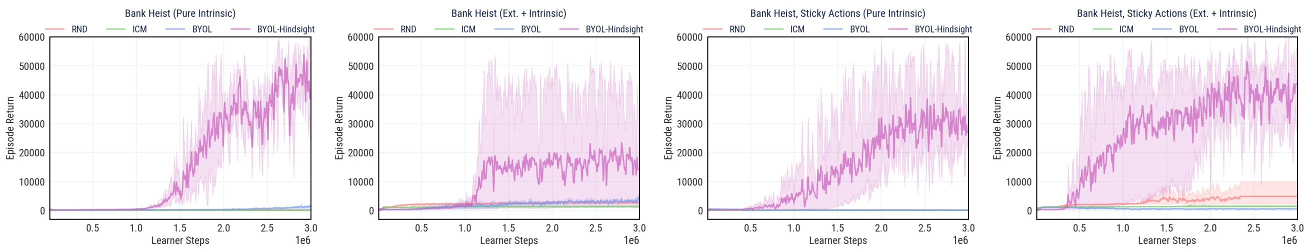
Figure 5. Pixel Noise.

[6] The composition of open-loop RNN and predictor/reconstructor networks is what we have abstractly denoted h_η in Examples 1–2.



(a) **Baseline: No Noise** (b) **Brownian Oscillators** (c) **Random Pixel Noise** (d) **On-Demand Pixel Noise**

Figure 6. Pycolab Maze, with Various Stochasticities. Performance measured by number of trackers touched in an episode (500-steps).



(a) **Natural Traps (Intrinsic Only)** (b) **Natural Traps (Ext. + Intrinsic)** (c) **Sticky Actions (Intrinsic Only)** (d) **Sticky Actions (Ext. + Intrinsic)**

Figure 7. Bank Heist, with Natural Traps and Sticky Actions. Performance measured by sum of extrinsic rewards obtained in an episode.

5.1. Pycolab Maze

First, to experiment with a variety of stochasticities in a controlled manner, we employ a Pycolab [76] maze (Figure 4): The agent spawns in the top right, and may explore past four (possibly stochastically oscillating) block elements (V1/2, H1/2), into the lower right where two coins are randomly spawned. The agent is purely intrinsically motivated, and progress is measured by trackers behind each of the block elements (R1–4). The agent only has access to a 5×5 frame (i.e. square radius 2) of its immediate surroundings as input.

Stochasticity We use four settings: "Baseline" (no noise); "Brownian Oscillators" (a form of state-dependent noise, where oscillators perform random walks along their axes of movements); "Random Pixel Noise" (a form of independent noise, which adds an extra layer of randomly sampled pixels to each frame with independent probability 0.25); as well as "On-Demand Pixel Noise" (a form of action-dependent noise, which does so whenever the no-op action is selected).

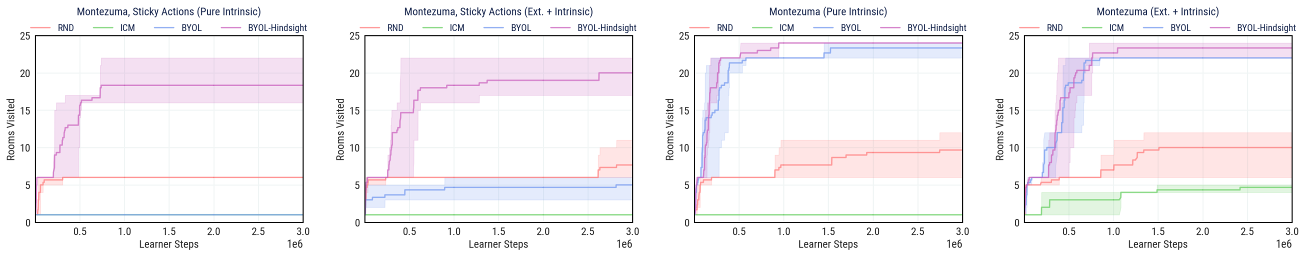
Results See Figure 6 (100k learner steps, 3 seeds). First, the "Baseline" setting tests non-specificity: Since there is no noise until the end, we expect curiosity-based exploration to do similarly with/without hindsight. For reference, we also show RND (in principle resilient to noise, as its targets are deterministic). All algorithms reach all four trackers (with RND eventually losing interest due to vanished rewards, as the environment is small). Second, in "Brownian Oscillators", `BYOL-Explore` fails to explore much beyond the first two trackers, as it is trapped by the unpredictable motion. In contrast, `BYOL-Hindsight` and RND both still explore the entire maze. Third, in "Random Pixel Noise" the results are similar, except both `BYOL-Explore` and RND do worse as the noise is an entire layer of random pixels (i.e. extremely diffuse), which outcompetes all other dynamics of the world in magnitude. Interestingly, while `BYOL-Hindsight` requires slightly longer to adapt, it still performs just as well. Lastly, the "On-Demand Pixel Noise" setting is most telling. `BYOL-Explore` is instantly trapped by the noise-inducing action, which it selects endlessly to generate intrinsic rewards. Even RND suffers greatly, which makes sense because the agent is no longer guaranteed a 0.75 probability of observing the world's unpolluted dynamics. In contrast, `BYOL-Hindsight` still performs as well as in the noise-free setting, underscoring robustness to different stochasticities.

5.2. Bank Heist

We use Atari with preprocessed grayscale 84×84-pixel images as input [77]. In all settings, we consider both "intrinsic-only" (no extrinsic signal) and "mixed" (extrinsic + intrinsic rewards) exploration regimes. In Bank Heist, the goal is to rob as many banks as possible while avoiding the police.

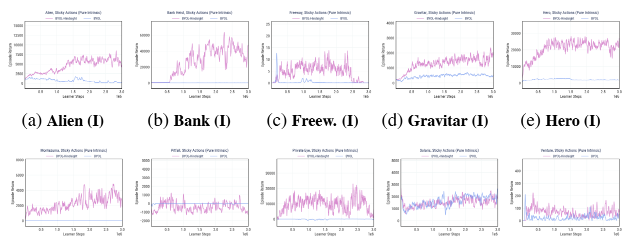
Stochasticity First, Bank Heist is characterized by naturally-occurring stochastic traps ("Natural Traps"), as noted by prior work [15]: It is impossible to predict where banks randomly regenerate and where bombs explode, thus a predictive error-based agent would simply endlessly enter and exit mazes while dropping bombs. Second, as an additional noise factor we use "Sticky Actions" [78] with stickiness 0.1.

Results See Figure 7 (3M learner steps, 3 seeds). Like in prior work, we measure the extrinsic reward per episode that the agent obtains, as a proxy for exploration ability. In both the "Natural Traps" and "Sticky Actions" settings, and in both intrinsic-only and mixed regimes, `BYOL-Explore`'s progress is immediately derailed by the stochastic traps, whereas `BYOL-Hindsight` achieves vastly better scores.



(a) **Sticky Actions (Intrinsic Only)** (b) **Sticky Actions (Ext. + Intrinsic)** (c) **Non-Sticky Baseline (Intrinsic Only)** (d) **Non-Sticky Baseline (Ext. + Intrinsic)**

Figure 8. Montezuma's Revenge, with Sticky Actions and Non-Sticky Baseline. Performance measured by rooms reached (episodic setting).



(a) **Alien (I)** (b) **Bank (I)** (c) **Freew. (I)** (d) **Gravitar (I)** (e) **Hero (I)**

(f) **Montez. (I)** (g) **Pitfall (I)** (h) **Priv. Eye (I)** (i) **Solaris (I)** (j) **Venture (I)**

Figure 9. Hard Exploration Games, with Sticky Actions. Performance measured by the sum of extrinsic rewards in an episode. Intrinsic-only "(I)" results shown; Figure 19 shows mixed "(E+I)".

5.3. Montezuma's Revenge

Playing Montezuma's Revenge requires learning complex dynamics including navigating around timed traps and moving enemies, and collecting keys to open doors in sequence.

Stochasticity First, Montezuma's Revenge is largely deterministic, which forms a natural baseline for testing non-specificity in a challenging exploration setting. Second, we use "Sticky Actions" similar to above to add stochasticity.

Results See Figure 8 (3M learner steps, 3 seeds). Like in prior work, exploration is measured by the number of different dungeon rooms the agent manages to discover over its lifetime. In "Sticky Actions", `BYOL-Explore` instantly flatlines in the intrinsic-only regime, and only does marginally better in the mixed regime. In contrast, `BYOL-Hindsight` explores most of the rooms in both regimes—which is an unprecedented result. In the "Non-Sticky Baseline", in both regimes `BYOL-Hindsight` does as well as the original performance of `BYOL-Explore`, which verifies non-specificity. See Appendix B.1 for evaluation results on episode return.

5.4. Hard Exploration Games

We conduct broad-based experiments for the ten hardest exploration games in Atari, where stochasticity is introduced with sticky actions as above. See Figure 9 for results in the intrinsic-only "(I)" regime, and also Figure 19 for the mixed "(E+I)" regime (3M learner steps, 1 seed). Like in prior work, we use extrinsic reward as a proxy for "interesting behavior". In the large majority of cases, `BYOL-Hindsight` improves over `BYOL-Explore`, especially when the latter simply flatlines. See Appendix B.7 for additional results and analysis.

5.5. Persistive Noise

While sticky actions is the standard protocol for adding stochasticity, we further design an especially challenging setting by corrupting observations with an additive layer of 84×84 pixel noise that persists across time as an action-triggerable random walk (Figure 5). See Appendix B.2 for full details and results, again verifying the robustness of the algorithm.

5.6. Temperature Sensitivity

The inner term in the contrastive loss (Objective 3) shares a similar form with contrastive losses in unsupervised representation learning [79–82], which admits a temperature parameter controlling the strength of penalties on negative samples [83]. See Appendix B.3 for a sensitivity analysis of the temperature parameter on the effectiveness of the algorithm.

5.7. Analysis of Invariance

For insight into invariance properties of the learned hindsight representations, see Appendix B.4 for analysis of intrinsic rewards over training; see Appendix B.5 for a comparison of prediction loss, reconstruction loss, and hindsight-only loss; and see Appendix B.6 for visualizations of what information the hindsight representations may be encoding.

6. Conclusion

In this work, we studied the problem that stochasticity poses to predictive error-based exploration. Theoretically, we refined our notion of curiosity to separate (learnable) epistemic knowledge from (unlearnable) aleatoric variation. Algorithmically, we proposed a method to learn (future-summarizing) representations of hindsight disentangled from (history-summarizing) representations of context. Practically, we arrived at a simple and scalable framework for generating (reducible) intrinsic rewards even in the presence of (irreducible) stochastic traps—without estimating the problematic entropy term at all. Our perspective shares connections with counterfactuals in policy evaluation [71–73], credit assignment [84–86], invariance [87–89], and fairness [90–93]. Future work may study explicitly generative world models to map stochastic latents to outcomes, as well as assessing the benefit of the approach in other stochastic domains such as NetHack. See Appendix D for an extended discussion.

References

[1] Nikolaus Kriegeskorte and Pamela K Douglas. Cognitive computational neuroscience. *Nature neuroscience*, 21(9):1148–1160, 2018.

[2] Tianpei Yang, Hongyao Tang, Chenjia Bai, Jinyi Liu, Jianye Hao, Zhaopeng Meng, and Peng Liu. Exploration in deep reinforcement learning: a comprehensive survey. *arXiv preprint arXiv:2109.06668*, 2021.

[3] Jürgen Schmidhuber. A possibility for implementing curiosity and boredom in model-building neural controllers. In *Proc. of the international conference on simulation of adaptive behavior: From animals to animats*, pages 222–227, 1991.

[4] Sebastian Thrun. Exploration in active learning. *Handbook of Brain Science and Neural Networks*, pages 381–384, 1995.

[5] Andrew G Barto, Satinder Singh, Nuttapong Chentanez, et al. Intrinsically motivated learning of hierarchical collections of skills. In *Proceedings of the 3rd International Conference on Development and Learning*, pages 112–19. Piscataway, NJ, 2004.

[6] Junhyuk Oh, Xiaoxiao Guo, Honglak Lee, Richard L Lewis, and Satinder Singh. Action-conditional video prediction using deep networks in atari games. *Advances in neural information processing systems*, 28, 2015.

[7] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. *Advances in neural information processing systems*, 29, 2016.

[8] Karol Gregor, Danilo Jimenez Rezende, Frederic Besse, Yan Wu, Hamza Merzic, and Aaron van den Oord. Shaping belief states with generative environment models for rl. *Advances in Neural Information Processing Systems*, 32, 2019.

[9] Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A Efros. Large-scale study of curiosity-driven learning. *International Conference on Learning Representations*, 2019.

[10] Bradly C Stadie, Sergey Levine, and Pieter Abbeel. Incentivizing exploration in reinforcement learning with deep predictive models. *International Conference on Learning Representations*, 2016.

[11] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pages 2778–2787. PMLR, 2017.

[12] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *International Conference on Learning Representations*, 2019.

[13] Hyoungseok Kim, Jaekyeom Kim, Yeonwoo Jeong, Sergey Levine, and Hyun Oh Song. Emi: Exploration with mutual information. In *International Conference on Machine Learning*, pages 3360–3369. PMLR, 2019.

[14] Zhaohan Daniel Guo, Shantanu Thakoor, Miruna Pîslar, Bernardo Avila Pires, Florent Altché, Corentin Tallec, Alaa Saade, Daniele Calandriello, Jean-Bastien Grill, Yunhao Tang, et al. Byol-explore: Exploration by bootstrapped prediction. *Advances in neural information processing systems*, 35, 2022.

[15] Augustine Mavor-Parker, Kimberly Young, Caswell Barry, and Lewis Griffin. How to stay curious while avoiding noisy tvs using aleatoric uncertainty estimation. In *International Conference on Machine Learning*, pages 15220–15240. PMLR, 2022.

[16] Leshem Choshen, Lior Fox, and Yonatan Loewenstein. Dora the explorer: Directed outreaching reinforcement action-selection. *International Conference on Learning Representations*, 2018.

[17] Laurent Orseau, Tor Lattimore, and Marcus Hutter. Universal knowledge-seeking agents for stochastic environments. In *International conference on algorithmic learning theory*, pages 158–172. Springer, 2013.

[18] Zhang-Wei Hong, Tsu-Jui Fu, Tzu-Yun Shann, and Chun-Yi Lee. Adversarial active exploration for inverse dynamics model learning. In *Conference on Robot Learning*, pages 552–565. PMLR, 2020.

[19] Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement. In *International conference on machine learning*, pages 5062–5071. PMLR, 2019.

[20] Kuno Kim, Megumi Sano, Julian De Freitas, Nick Haber, and Daniel Yamins. Active world model learning with progress curiosity. In *International conference on machine learning*, pages 5306–5315. PMLR, 2020.

[21] Mikael Henaff. Explicit explore-exploit algorithms in continuous state spaces. *Advances in Neural Information Processing Systems*, 32, 2019.

[22] Pranav Shyam, Wojciech Jaśkowski, and Faustino Gomez. Model-based active exploration. In *International conference on machine learning*, pages 5779–5788. PMLR, 2019.

[23] Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. *Advances in neural information processing systems*, 29, 2016.

[24] Ian Osband, John Aslanides, and Albin Cassirer. Randomized prior functions for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018.

[25] Alexander L Strehl and Michael L Littman. An analysis of model-based interval estimation for markov decision processes. *Journal of Computer and System Sciences*, 74, 2008.

[26] Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. # exploration: A study of count-based exploration for deep reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

[27] Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. *Advances in neural information processing systems*, 29, 2016.

[28] Georg Ostrovski, Marc G Bellemare, Aäron Oord, and Rémi Munos. Count-based exploration with neural density models. In *International conference on machine learning*, pages 2721–2730. PMLR, 2017.

[29] Rui Zhao and Volker Tresp. Curiosity-driven experience prioritization via density estimation. *Advances in neural information processing systems*, 31, 2018.

[30] Omar Darwiche Domingues, Corentin Tallec, Remi Munos, and Michal Valko. Density-based bonuses on learned representations for reward-free exploration in deep reinforcement learning. In *ICML 2021 Workshop on Unsupervised Reinforcement Learning*, 2021.

[31] Justin Fu, John Co-Reyes, and Sergey Levine. Ex2: Exploration with exemplar models for deep reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

[32] Yannis Flet-Berliac, Johan Ferret, Olivier Pietquin, Philippe Preux, and Matthieu Geist. Adversarially guided actor-critic. *International Conference on Learning Representations*, 2021.

[33] Nikolay Savinov, Anton Raichuk, Raphaël Marinier, Damien Vincent, Marc Pollefeys, Timothy Lillicrap, and Sylvain Gelly. Episodic curiosity through reachability. *International Conference on Learning Representations*, 2019.

[34] Adrià Puigdomènech Badia, Pablo Sprechmann, Alex Vitvitskyi, Daniel Guo, Bilal Piot, Steven Kapturowski, Olivier Tieleman, Martín Arjovsky, Alexander Pritzel, Andew Bolt, et al. Never give up: Learning directed exploration strategies. *International Conference on Learning Representations*, 2020.

[35] Adrià Puigdomènech Badia, Bilal Piot, Steven Kapturowski, Pablo Sprechmann, Alex Vitvitskyi, Zhaohan Daniel Guo, and Charles Blundell. Agent57: Outperforming the atari human benchmark. In *International Conference on Machine Learning*, pages 507–517. PMLR, 2020.

[36] Marlos C Machado, Marc G Bellemare, and Michael Bowling. Count-based exploration with the successor representation. In *ICML Workshop on Exploration in Reinforcement Learning*, 2018.

[37] Min-hwan Oh and Garud Iyengar. Directed exploration in pac model-free reinforcement learning. In *ICML Workshop on Exploration in Reinforcement Learning*, 2018.

[38] Marlos C Machado, Marc G Bellemare, and Michael Bowling. Count-based exploration with the successor representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[39] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145, 1996.

[40] Laurent Itti and Pierre Baldi. Bayesian surprise attracts human attention. *Vision research*, 49(10):1295–1306, 2009.

[41] Mauricio Araya, Olivier Buffet, Vincent Thomas, and Françcois Charpillet. A pomdp extension with belief-dependent rewards. *Advances in neural information processing systems*, 23, 2010.

[42] Yi Sun, Faustino Gomez, and Jürgen Schmidhuber. Planning to be surprised: Optimal bayesian exploration in dynamic environments. In *International conference on artificial general intelligence*, pages 41–51. Springer, 2011.

[43] Susanne Still and Doina Precup. An information-theoretic approach to curiosity-driven reinforcement learning. *Theory in Biosciences*, 131(3):139–148, 2012.

[44] Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. *Advances in neural information processing systems*, 29, 2016.

[45] Ramanan Sekar, Oleh Rybkin, Kostas Daniilidis, Pieter Abbeel, Danijar Hafner, and Deepak Pathak. Planning to explore via self-supervised world models. In *International Conference on Machine Learning*, pages 8583–8592. PMLR, 2020.

[46] Russell Mendonca, Oleh Rybkin, Kostas Daniilidis, Danijar Hafner, and Deepak Pathak. Discovering and achieving goals via world models. *Advances in Neural Information Processing Systems*, 34:24379–24391, 2021.

[47] Jürgen Schmidhuber. Curious model-building control systems. In *Proc. international joint conference on neural networks*, pages 1458–1463, 1991.

[48] Pierre-Yves Oudeyer, Frdric Kaplan, and Verena V Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE transactions on evolutionary computation*, 11(2):265–286, 2007.

[49] Mohammad Gheshlaghi Azar, Bilal Piot, Bernardo Avila Pires, Jean-Bastien Grill, Florent Altché, and Rémi Munos. World discovery models. *arXiv preprint arXiv:1902.07685*, 2019.

[50] Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, pages 2681–2691. PMLR, 2019.

[51] Hao Liu and Pieter Abbeel. Behavior from the void: Unsupervised active pre-training. *Advances in Neural Information Processing Systems*, 34:18459–18473, 2021.

[52] Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Alaa Saade, Shantanu Thakoor, Bilal Piot, Bernardo Avila Pires, Michal Valko, Thomas Mesnard, Tor Lattimore, and Rémi Munos. Geometric entropic exploration. *arXiv preprint arXiv:2101.02055*, 2021.

[53] Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Reinforcement learning with prototypical representations. In *International Conference on Machine Learning*, pages 11920–11931. PMLR, 2021.

[54] Karol Gregor, Danilo Jimenez Rezende, and Daan Wierstra. Variational intrinsic control. *International Conference on Learning Representations*, 2017.

[55] Joshua Achiam, Harrison Edwards, Dario Amodei, and Pieter Abbeel. Variational option discovery algorithms. *arXiv preprint arXiv:1807.10299*, 2018.

[56] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. *International Conference on Learning Representations*, 2019.

[57] Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric Xing, Sergey Levine, and Ruslan Salakhutdinov. Efficient exploration via state marginal matching. *arXiv preprint arXiv:1906.05274*, 2019.

[58] Víctor Campos, Alexander Trott, Caiming Xiong, Richard Socher, Xavier Giró-i Nieto, and Jordi Torres. Explore, discover and learn: Unsupervised discovery of state-covering skills. In *International Conference on Machine Learning*, pages 1317–1327. PMLR, 2020.

[59] Archit Sharma, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. Dynamics-aware unsupervised discovery of skills. *International Conference on Learning Representations*, 2020.

[60] Kate Baumli, David Warde-Farley, Steven Hansen, and Volodymyr Mnih. Relative variational intrinsic control. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6732–6740, 2021.

[61] Oliver Groth, Markus Wulfmeier, Giulia Vezzani, Vibhavari Dasagi, Tim Hertweck, Roland Hafner, Nicolas Heess, and Martin Riedmiller. Is curiosity all you need? on the utility of emergent behaviours from curious exploration. *arXiv preprint arXiv:2109.08603*, 2021.

[62] Taehwan Kwon. Variational intrinsic control revisited. *International Conference on Learning Representations*, 2022.

[63] Hao Liu and Pieter Abbeel. Aps: Active pretraining with successor features. In *International Conference on Machine Learning*, pages 6736–6747. PMLR, 2021.

[64] Benjamin Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. The information geometry of unsupervised reinforcement learning. *International Conference on Learning Representations*, 2022.

[65] Michael Laskin, Hao Liu, Xue Bin Peng, Denis Yarats, Aravind Rajeswaran, and Pieter Abbeel. Cic: Contrastive intrinsic control for unsupervised skill discovery. *arXiv preprint arXiv:2202.00161*, 2022.

[66] Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

[67] Carlos Florensa, David Held, Xinyang Geng, and Pieter Abbeel. Automatic goal generation for reinforcement learning agents. In *International conference on machine learning*, pages 1515–1528. PMLR, 2018.

[68] Ashvin V Nair, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine. Visual reinforcement learning with imagined goals. *Advances in neural information processing systems*, 31, 2018.

[69] Meng Fang, Tianyi Zhou, Yali Du, Lei Han, and Zhengyou Zhang. Curriculum-guided hindsight experience replay. *Advances in neural information processing systems*, 32, 2019.

[70] Yunzhi Zhang, Pieter Abbeel, and Lerrel Pinto. Automatic curriculum learning through value disagreement. *Advances in Neural Information Processing Systems*, 33:7648–7659, 2020.

[71] Lars Buesing, Theophane Weber, Yori Zwols, Sebastien Racaniere, Arthur Guez, Jean-Baptiste Lespiau, and Nicolas Heess. Woulda, coulda, shoulda: Counterfactually-guided policy search. *arXiv preprint arXiv:1811.06272*, 2018.

[72] Michael Oberst and David Sontag. Counterfactual off-policy evaluation with gumbel-max structural causal models. In *International Conference on Machine Learning*, pages 4881–4890. PMLR, 2019.

[73] Guy Lorberbom, Daniel D Johnson, Chris J Maddison, Daniel Tarlow, and Tamir Hazan. Learning generalized gumbel-max causal mechanisms. *Advances in Neural Information Processing Systems*, 34:26792–26803, 2021.

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

[75] H Francis Song, Abbas Abdolmaleki, Jost Tobias Springenberg, Aidan Clark, Hubert Soyer, Jack W Rae, Seb Noury, Arun Ahuja, Siqi Liu, Dhruva Tirumala, et al. V-mpo: On-policy maximum a posteriori policy optimization for discrete and continuous control. *arXiv preprint arXiv:1909.12238*, 2019.

[76] Thomas Stepleton. The pycolab game engine, 2017. *URL https://github. com/deepmind/pycolab*, 2017.

[77] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

[78] Marlos C Machado, Marc G Bellemare, Erik Talvitie, Joel Veness, Matthew Hausknecht, and Michael Bowling. Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. *Journal of Artificial Intelligence Research*, 61:523–562, 2018.

[79] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[80] Michael Tschannen, Josip Djolonga, Paul K Rubenstein, Sylvain Gelly, and Mario Lucic. On mutual information maximization for representation learning. *arXiv preprint arXiv:1907.13625*, 2019.

[81] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[82] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[83] Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504, 2021.

[84] Anna Harutyunyan, Will Dabney, Thomas Mesnard, Mohammad Gheshlaghi Azar, Bilal Piot, Nicolas Heess, Hado P van Hasselt, Gregory Wayne, Satinder Singh, Doina Precup, et al. Hindsight credit assignment. *Advances in neural information processing systems*, 32, 2019.

[85] Chris Nota, Philip Thomas, and Bruno C Da Silva. Posterior value functions: Hindsight baselines for policy gradient methods. In *International Conference on Machine Learning*, pages 8238–8247. PMLR, 2021.

[86] Thomas Mesnard, Théophane Weber, Fabio Viola, Shantanu Thakoor, Alaa Saade, Anna Harutyunyan, Will Dabney, Tom Stepleton, Nicolas Heess, Arthur Guez, et al. Counterfactual credit assignment in model-free reinforcement learning. In *International Conference on Machine Learning*, 2021.

[87] Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. *arXiv preprint arXiv:2006.10742*, 2020.

[88] Ioana Bica, Daniel Jarrett, and Mihaela van der Schaar. Invariant causal imitation learning for generalizable policies. *Advances in Neural Information Processing Systems*, 34:3952–3964, 2021.

[89] Chaochao Lu, José Miguel Hernández-Lobato, and Bernhard Schölkopf. Invariant causal representation learning for generalization in imitation and reinforcement learning. In *ICLR 2022*, 2022.

[90] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. The variational fair autoencoder. *International Conference on Learning Representations*, 2016.

[91] Daniel Moyer, Shuyang Gao, Rob Brekelmans, Aram Galstyan, and Greg Ver Steeg. Invariant representations without adversarial training. *Advances in Neural Information Processing Systems*, 31, 2018.

[92] Mohammad Lotfollahi, Mohsen Naghipourfar, Fabian J Theis, and F Alexander Wolf. Conditional out-of-sample generation for unpaired data using trvae. *arXiv preprint arXiv:1910.01791*, 2019.

[93] Adam Foster, Árpi Vezér, Craig A Glastonbury, Páidí Creed, Samer Abujudeh, and Aaron Sim. Contrastive mixture of posteriors. In *International Conference on Machine Learning*, pages 6578–6621. PMLR, 2022.

[94] David Barber Felix Agakov. The im algorithm: a variational approach to information maximization. *Advances in neural information processing systems*, 16(320):201, 2004.

[95] Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180. PMLR, 2019.

[96] Matteo Hessel, Hubert Soyer, Lasse Espeholt, Wojciech Czarnecki, Simon Schmitt, and Hado van Hasselt. Multi-task deep reinforcement learning with popart. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

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

[98] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[99] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[100] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[101] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *AAAI conference on artificial intelligence*, 2016.

[102] Daniel Jarrett, Ioana Bica, and Mihaela van der Schaar. Time-series generation by contrastive imitation. *Advances in Neural Information Processing Systems*, 34:28968–28982, 2021.

[103] Danijar Hafner. Benchmarking the spectrum of agent capabilities. *arXiv preprint arXiv:2109.06780*, 2021.

[104] Mikael Henaff, Roberta Raileanu, Minqi Jiang, and Tim Rocktäschel. Exploration via elliptical episodic bonuses. *arXiv preprint arXiv:2210.05805*, 2022.

[105] Alaa Saade, Steven Kapturowski, Daniele Calandriello, Charles Blundell, Michal Valko, Pablo Sprechmann, and Bilal Piot. Robust exploration via clustering-based online density estimation. 2023.

[106] Bilal Piot, Zhaohan Daniel Guo, Shantanu Thakoor, and Mohammad Gheshlaghi Azar. Blade: Robust exploration via diffusion models. In *Deep Reinforcement Learning Workshop NeurIPS 2022*, 2022.

[107] Yunhao Tang, Zhaohan Daniel Guo, Pierre Harvey Richemond, Bernardo Ávila Pires, Yash Chandak, Rémi Munos, Mark Rowland, Mohammad Gheshlaghi Azar, Charline Le Lan, Clare Lyle, et al. Understanding self-predictive learning for reinforcement learning. *arXiv preprint arXiv:2212.03319*, 2022.

[108] Mengjiao Yang, Dale Schuurmans, Pieter Abbeel, and Ofir Nachum. Dichotomy of control. *arXiv preprint arXiv:2210.13435*, 2022.

Curiosity in Hindsight: Intrinsic Exploration in Stochastic Environments

Daniel Jarrett [1] **Corentin Tallec** [1] **Florent Altché** [1] **Thomas Mesnard** [1] **Rémi Munos** [1] **Michal Valko** [1]

Supplementary Materials

[1]DeepMind. Correspondence to: Dan Jarrett <jarrettd@google.com>.

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

A. Proofs of Propositions

To simplify our notation, we remove subscripts such that X, A, Y denotes the transition X_t, A_t, X_{t+1}, and Z denotes the latent Z_{t+1}. Then the environment's dynamics is given by $\tau(Y|x,a)$, the agent's policy is given by $\pi(A|x)$, and the induced state visitation is given by $\rho_\pi(X)$. The generator is denoted $p_\theta(Z|x,a,y)$, reconstructor $f_\eta(x,a,z)$, and critic $g_\nu(x,a,z)$.

A.1. Pointwise Mutual Information

We start by deriving several lemmas that will be useful, the first being a pointwise version of Barber and Agakov's variational lower bound on mutual information [94, 95]:

Lemma 3 (Pointwise Barber-Agakov) Denote the pointwise mutual information:

$$\text{PMI}_\theta(x,a;z) \coloneqq \log \frac{p_\theta(z|x,a)}{p_\theta(z)} \tag{25}$$

Then for any variational distribution q:

$$\mathbb{E}_{Z\sim p_\theta(\cdot|x,a)}\text{PMI}_\theta(x,a;Z) \geq \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \log \frac{q(Z|x,a)}{p_\theta(Z)} \tag{26}$$

Proof. Starting from the left hand side:

$$\begin{aligned}
\mathbb{E}_{Z\sim p_\theta(\cdot|x,a)}\text{PMI}_\theta(x,a;Z) &= \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \log \frac{p_\theta(Z|x,a)}{p_\theta(Z)} && (27)\\
&= \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \log \frac{p_\theta(Z|x,a)}{p_\theta(Z)} + \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \log \frac{q(Z|x,a)}{q(Z|x,a)} && (28)\\
&= \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \log \frac{q(Z|x,a)}{p_\theta(Z)} + D_{\text{KL}}\big(p_\theta(Z|x,a)\|q(Z|x,a)\big) && (29)\\
&\geq \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \log \frac{q(Z|x,a)}{p_\theta(Z)} && (30)
\end{aligned}$$

which completes the proof. □

Next, we define a generic contrastive expression with $K-1$ "negative" samples of Z, and show that taking its expectation with respect to those samples yields a valid (i.e. normalized) probability density:

Lemma 4 (Normalized Variational) Given independent samples $z_{1:K-1}$ from p_θ, define:

$$q(z|x,a,z_{1:K-1}) \coloneqq \frac{p_\theta(z)\cdot e^{g_\nu(x,a,z)}}{\frac{1}{K}\left(e^{g_\nu(x,a,z)} + \sum_{i=1}^{K-1} e^{g_\nu(x,a,z_i)}\right)} \tag{31}$$

then the following defines a normalized density:

$$q(Z|x,a) \coloneqq \mathbb{E}_{Z_{1:K-1}\sim p_\theta^{K-1}} q(Z|x,a,Z_{1:K-1}) \tag{32}$$

Proof. The expectation integrates to one:

$$\begin{aligned}
\int_{\mathcal{Z}} q(z|x,a)dz &= \int_{\mathcal{Z}} \mathbb{E}_{Z_{1:K-1}\sim p_\theta^{K-1}} \frac{p_\theta(z)\cdot e^{g_\nu(x,a,z)}}{\frac{1}{K}\left(e^{g_\nu(x,a,z)} + \sum_{i=1}^{K-1} e^{g_\nu(x,a,Z_i)}\right)} dz && (33)\\
&= \mathbb{E}_{\substack{Z\sim p_\theta \\ Z_{1:K-1}\sim p_\theta^{K-1}}} \frac{e^{g_\nu(x,a,Z)}}{\frac{1}{K}\left(e^{g_\nu(x,a,Z)} + \sum_{i=1}^{K-1} e^{g_\nu(x,a,Z_i)}\right)} && (34)
\end{aligned}$$

$$= K \cdot \mathbb{E}_{Z_{1:K} \sim p_\theta^K} \frac{e^{g_\nu(x,a,Z_1)}}{\sum_{i=1}^K e^{g_\nu(x,a,Z_i)}} \tag{35}$$

$$= \mathbb{E}_{Z_{1:K} \sim p_\theta^K} \frac{\sum_{j=1}^K e^{g_\nu(x,a,Z_j)}}{\sum_{i=1}^K e^{g_\nu(x,a,Z_i)}} \tag{36}$$

$$= 1 \tag{37}$$

which completes the proof. □

These two results allow us to show that the information Z contains on a tuple x, a—with respect to the generator parameterized as θ—is lower-bounded by the x, a-conditioned contrastive loss between "positive" samples $Z \sim p_\theta(\cdot|x, a)$ from the posterior and "negative" samples $Z \sim p_\theta$ from the prior:

Lemma 5 (State-Action Lower Bound) The x, a-wise mutual information satisfies:

$$\begin{aligned}
&\mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \mathrm{PMI}_\theta(x, a; Z) \\
&\qquad \geq \mathbb{E}_{\substack{Z\sim p_\theta(\cdot|x,a) \\ Z_{1:K-1}\sim p_\theta^{K-1}}} \log \frac{e^{g_\nu(x,a,Z)}}{\frac{1}{K}\left(e^{g_\nu(x,a,Z)} + \sum_{i=1}^{K-1} e^{g_\nu(x,a,Z_i)}\right)}
\end{aligned} \tag{38}$$

Proof. Use Lemmas 3 and 4, then Jensen's inequality:

$$\mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \mathrm{PMI}_\theta(x, a; Z)$$

$$\geq \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \log \frac{q(Z|x, a)}{p_\theta(Z)} \tag{39}$$

$$= \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \log \mathbb{E}_{Z_{1:K-1}\sim p_\theta^{K-1}} \frac{q(Z|x, a, Z_{1:K-1})}{p_\theta(Z)} \tag{40}$$

$$\geq \mathbb{E}_{\substack{Z\sim p_\theta(\cdot|x,a) \\ Z_{1:K-1}\sim p_\theta^{K-1}}} \log \frac{q(Z|x, a, Z_{1:K-1})}{p_\theta(Z)} \tag{41}$$

$$= \mathbb{E}_{\substack{Z\sim p_\theta(\cdot|x,a) \\ Z_{1:K-1}\sim p_\theta^{K-1}}} \log \frac{e^{g_\nu(x,a,Z)}}{\frac{1}{K}\left(e^{g_\nu(x,a,Z)} + \sum_{i=1}^{K-1} e^{g_\nu(x,a,Z_i)}\right)} \tag{42}$$

which completes the proof. □

Next, we show that our invariance loss (Objective 2) for a tuple x, a, z is equal to the pointwise mutual information in the limit of infinitely large negative batches, assuming an optimal critic parameter:

Lemma 6 (Pointwise Asymptotic Equality) Define the transition-wise contrastive loss:

$$\mathcal{L}_{\theta,\nu}^{K,\text{con.}}(x, a, z) \coloneqq \mathbb{E}_{Z_{1:K-1}\sim p_\theta^{K-1}} \log \frac{e^{g_\nu(x,a,z)}}{\frac{1}{K}\left(e^{g_\nu(x,a,z)} + \sum_{i=1}^{K-1} e^{g_\nu(x,a,Z_i)}\right)} \tag{43}$$

and the optimal critic parameter:

$$\nu_K^* \coloneqq \arg\max_\nu \mathbb{E}_{\substack{X\sim\rho_\pi \\ A\sim\pi(\cdot|X) \\ Y\sim\tau(\cdot|X,A) \\ Z\sim p_\theta(\cdot|X,A,Y)}} \mathcal{L}_{\theta,\nu}^{K,\text{con.}}(X, A, Z) \tag{44}$$

Then $\lim_{K\to\infty} \mathcal{L}_{\theta,\nu_K^*}^{K,\text{con.}}(x, a, z) = \mathrm{PMI}_\theta(x, a; z)$.

Proof. The $\mathbb{E}[\mathcal{L}_{\theta,\nu}^{K,\text{con.}}(X, A, Z)]$ term is just the InfoNCE loss between variables Z and X, A, so we know that ν_K^* satisfies $g_{\nu_K^*}(x, a, z) = \log \frac{p_\theta(z|x,a)}{p_\theta(z)} + c(x, a)$. Substituting this back into $\mathcal{L}_{\theta,\nu}^{K,\text{con.}}(x, a, z)$:

$$\lim_{K\to\infty} \mathcal{L}^{K,\text{con.}}_{\theta,\nu^*_K}(x,a,z) \tag{45}$$

$$= \lim_{K\to\infty} \mathbb{E}_{Z_{1:K-1}\sim p_\theta^{K-1}} \log \frac{e^{g_{\nu^*_K}(x,a,z)}}{\frac{1}{K}\left(e^{g_{\nu^*_K}(x,a,z)} + \sum_{i=1}^{K-1} e^{g_{\nu^*_K}(x,a,Z_i)}\right)} \tag{46}$$

$$= \lim_{K\to\infty} \mathbb{E}_{Z_{1:K-1}\sim p_\theta^{K-1}} \log \frac{\frac{p_\theta(z|x,a)}{p_\theta(z)}}{\frac{1}{K}\left(\frac{p_\theta(z|x,a)}{p_\theta(z)} + \sum_{i=1}^{K-1} \frac{p_\theta(Z_i|x,a)}{p_\theta(Z_i)}\right)} \tag{47}$$

$$= \lim_{K\to\infty} \mathbb{E}_{Z_{1:K-1}\sim p_\theta^{K-1}} \left[\log \frac{p_\theta(z|x,a)}{p_\theta(z)} - \log \frac{\frac{p_\theta(z|x,a)}{p_\theta(z)} + \sum_{i=1}^{K-1} \frac{p_\theta(Z_i|x,a)}{p_\theta(Z_i)}}{K}\right] \tag{48}$$

$$= \log \frac{p_\theta(z|x,a)}{p_\theta(z)} - \lim_{K\to\infty} \log \frac{\frac{p_\theta(z|x,a)}{p_\theta(z)} + K - 1}{K} \tag{49}$$

$$= \mathrm{PMI}_\theta(x,a;z) \tag{50}$$

which completes the proof. □

Lastly, recall the following basic relationship:

Lemma 7 (Conditional Mutual Information) Conditioned on any x, a, we have that:

$$\mathbb{I}_\theta[Y;Z|x,a] = \mathbb{H}[Y|x,a] + \mathbb{H}_\theta[Y|x,a,Z] \tag{51}$$

Proof. Starting from the left hand side:

$$\mathbb{I}_\theta[Y;Z|x,a] \coloneqq \mathbb{E}_{Z\sim p_\theta} D_{\mathrm{KL}}\big(p_\theta(Y|x,a,Z)\|\tau(Y|x,a)\big) \tag{52}$$

$$= \mathbb{E}_{\substack{Z\sim p_\theta \\ Y\sim p_\theta(\cdot|x,a,Z)}} \log p_\theta(Y|x,a,Z) - \mathbb{E}_{\substack{Z\sim p_\theta \\ Y\sim p_\theta(\cdot|x,a,Z)}} \tau(Y|x,a) \tag{53}$$

$$= -\int_{\mathcal{Z}} p_\theta(z)\mathbb{H}_\theta[Y|x,a,z]dz - \mathbb{E}_{\substack{Y\sim\tau(\cdot|x,a) \\ Z\sim p_\theta(\cdot|x,a,Y)}} \tau(Y|x,a) \tag{54}$$

$$= \mathbb{H}[Y|x,a] - \mathbb{H}_\theta[Y|x,a,Z] \tag{55}$$

which completes the proof. □

A.2. Optimistic Exploration

In our structural causal model, by construction Z captures all sources of noise—that is, there is no residual noise in each outcome Y. However, for the purposes of optimization, while learning η we let the residual error be captured by a Gaussian "log-likelihood" (note that λ plays the role of "$2\sigma^2$"):

$$\log p_\eta(Y|x,a,z) \coloneqq -\frac{1}{2}\log(\lambda\pi) - \frac{1}{\lambda}\big(Y - f_\eta(x,a,z)\big)^2 \tag{56}$$

and note that θ also induces a log-likelihood of the "ground-truth" conditional:

$$\log p_\theta(Y|x,a,z) \coloneqq \log \frac{p_\theta(z|x,a,Y)\tau(Y|x,a)\pi(a,x)\rho_\pi(x)}{\int_{\mathcal{Y}} p_\theta(z|x,a,y)\tau(y|x,a)\pi(a|x)\rho_\pi(x)dy} \tag{57}$$

Now, recall the reconstruction loss and (state-action) reconstruction bonus:

$$\mathcal{L}^{\text{rec.}}_\eta(x,a,z,y) \coloneqq \big\|y - f_\eta(x,a,z)\big\|_2^2 \tag{58}$$

$$\mathcal{R}^{\text{rec.}}_{\theta,\eta}(x,a) \coloneqq \mathbb{E}_{\substack{Y\sim\tau(\cdot|x,a) \\ Z\sim p_\theta(\cdot|x,a,Y)}} \mathcal{L}^{\text{rec.}}_\eta(x,a,Z,Y) \tag{59}$$

as well as the invariance loss and (state-action) invariance bonus:

$$\mathcal{L}_\theta^{\text{inv.}}(x,a,z) \coloneqq \text{PMI}_\theta(x,a;z) \tag{60}$$

$$\mathcal{R}_\theta^{\text{inv.}}(x,a) \coloneqq \mathbb{E}_{\substack{Y\sim\tau(\cdot|x,a)\\ Z\sim p_\theta(\cdot|x,a,Y)}} \mathcal{L}_\theta^{\text{inv.}}(x,a,Z) \tag{61}$$

We now show that Theorem 1 is true, which we restate using our subscript-less notation:

Theorem 8 (Optimistic Exploration) Let λ satisfy the inequality $\frac{1}{2}\log(\lambda\pi) \leq \mathbb{H}_\theta[Y|x,a,Z] + D_{\text{KL}}\big(p_\theta(Z|x,a)\|p_\theta(Z)\big)$, where π denotes here the mathematical constant (not the agent's policy). Then:

$$\mathcal{R}_{\theta,\eta}(x,a) \geq D_{\text{KL}}\big(\tau(Y|x,a)\|\tau_{\theta,\eta}(Y|x,a)\big) \tag{62}$$

where $\tau_{\theta,\eta}(Y|x,a) \coloneqq \mathbb{E}_{Z\sim p_\theta} p_\eta(Y|x,a,Z)$ denotes the learned world model. Furthermore, assuming realizability, rewards vanish at optimal parameters θ^*, η^*:

$$\mathcal{R}_{\theta^*,\eta^*}(x,a) = 0 \quad \forall x,a \in \text{supp}(\rho_\pi) \tag{63}$$

Proof. Use Definition 2, then the constraint on λ, then Lemma 7:

$$\begin{aligned}
\mathcal{R}_{\theta,\eta}(x,a) &\coloneqq \frac{1}{\lambda}\mathcal{R}_{\theta,\eta}^{\text{rec.}}(x,a) + \mathcal{R}_\theta^{\text{inv.}}(x,a) && (64)\\
&= \mathbb{E}_{\substack{Y\sim\tau(\cdot|x,a)\\ Z\sim p_\theta(\cdot|x,a,Y)}} \frac{1}{\lambda}\big(Y - f_\eta(x,a,Z)\big)^2 + \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)}\text{PMI}_\theta(x,a;Z) && (65)\\
&= \mathbb{E}_{\substack{Y\sim\tau(\cdot|x,a)\\ Z\sim p_\theta(\cdot|x,a,Y)}} \frac{1}{\lambda}\big(Y - f_\eta(x,a,Z)\big)^2 + D_{\text{KL}}\big(p_\theta(Z|x,a)\|p_\theta(Z)\big) && (66)\\
&\geq -\mathbb{E}_{\substack{Y\sim\tau(\cdot|x,a)\\ Z\sim p_\theta(\cdot|x,a,Y)}} \log p_\eta(Y|x,a,Z) - \mathbb{H}_\theta[Y|x,a,Z] && (67)\\
&= -\mathbb{E}_{\substack{Y\sim\tau(\cdot|x,a)\\ Z\sim p_\theta(\cdot|x,a,Y)}} \log p_\eta(Y|x,a,Z) + \mathbb{I}_\theta[Y;Z|x,a] - \mathbb{H}[Y|x,a] && (68)\\
&= -\mathbb{E}_{\substack{Y\sim\tau(\cdot|x,a)\\ Z\sim p_\theta(\cdot|x,a,Y)}} \log p_\eta(Y|x,a,Z) \leftarrow \textbf{remaining}\text{ stochasticity}\\
&\quad + \mathbb{E}_{Y\sim\tau(\cdot|x,a)} D_{\text{KL}}\big(p_\theta(Z|x,a,Y)\|p_\theta(Z|x,a)\big) \leftarrow \textbf{hindsight}\text{ information}\\
&\quad - \mathbb{E}_{Y\sim\tau(\cdot|x,a)}\big[-\log\tau(Y|x,a)\big] \leftarrow \textbf{total}\text{ stochasticity} && (69)\\
&\geq -\mathbb{E}_{Y\sim\tau(\cdot|x,a)}\big[\,\mathbb{E}_{Z\sim p_\theta(\cdot|x,a,Y)}\log p_\eta(Y|x,a,Z)\\
&\quad - D_{\text{KL}}\big(p_\theta(Z|x,a,Y)\|p_\theta(Z|x,a)\big) + D_{\text{KL}}\big(p_\theta(Z|x,a,Y)\|p_\eta(Z|x,a,Y)\big)\big]\\
&\quad + \mathbb{E}_{Y\sim\tau(\cdot|x,a)}\log\tau(Y|x,a) && (70)\\
&= -\mathbb{E}_{Y\sim\tau(\cdot|x,a)}\log\mathbb{E}_{Z\sim p_\theta} p_\eta(Y|x,a,Z) + \mathbb{E}_{Y\sim\tau(\cdot|x,a)}\log\tau(Y|x,a) && (71)\\
&= -\mathbb{E}_{Y\sim\tau(\cdot|x,a)}\log\tau_{\theta,\eta}(Y|x,a) + \mathbb{E}_{Y\sim\tau(\cdot|x,a)}\log\tau(Y|x,a) && (72)\\
&= D_{\text{KL}}\big(\tau(Y|x,a)\|\tau_{\theta,\eta}(Y|x,a)\big) && (73)
\end{aligned}$$

For the second part, we want to show that for the objective:

$$J(\theta,\eta;\lambda) \coloneqq \mathbb{E}_{\substack{X\sim\rho_\pi\\ A\sim\pi(\cdot|X)}}\left[\frac{1}{\lambda}\mathcal{R}_{\theta,\eta}^{\text{rec.}}(X,A) + \mathcal{R}_\theta^{\text{inv.}}(X,A)\right] \tag{74}$$

its optimal value is zero:

$$\min_{\theta,\eta} J(\theta, \eta; \lambda) = 0 \tag{75}$$

Take any MDP. By reparameterization, we know that there exists an equivalent graphical representation under which Z is exogenous. Assuming realizability, let η^* be such that $f_{\eta^*} = f$, and let θ^* be such that $p_{\theta^*}(Z|x, a, y) = p_{\eta^*}(Z|x, a, y)$ for any x, a, y. First, by construction we have that $Z \perp X, A$, so the mutual information between Z and X, A must be zero:

$$\mathbb{E}_{\substack{X\sim\rho_\pi \\ A\sim\pi(\cdot|X)}} \mathcal{R}_\theta^{\text{inv.}}(X, A) = \mathbb{E}_{\substack{X\sim\rho_\pi \\ A\sim\pi(\cdot|X) \\ Z\sim p_\theta(\cdot|X,A)}} \text{PMI}_\theta(X, A; Z) \tag{76}$$

$$= \mathbb{I}_\theta[X, A; Z] \tag{77}$$

$$= 0 \tag{78}$$

Second, by consistency of counterfactuals $f_{\eta^*}(x, a, Z) = y$ for any $Z \sim p_{\theta^*}(\cdot|x, a, y)$, so the reconstruction term is also zero, which completes the proof. □

The intuition is as follows: Assuming realizability, at convergence "hindsight information" and "total stochasticity" cancel (i.e. neither more nor less), and the "remaining stochasticity" term goes to zero.

A.3. Optimal Invariance

We now show that Theorem 2 is true, which we restate using our subscript-less notation:

Theorem 9 (Optimal Invariance) The contrastive bonus lower-bounds the (ideal) invariance bonus for any pair x, a:

$$\mathcal{R}_{\theta,\nu}^{K,\text{con.}}(x, a) \le \mathcal{R}_\theta^{\text{inv.}}(x, a) \tag{79}$$

Furthermore, assuming realizability, for optimal critic parameter $\nu_K^* := \arg\max_\nu \mathbb{E}_{X,A\sim\rho_\pi} \mathcal{R}_{\theta,\nu}^{K,\text{con.}}(X, A)$ the bound is asymptotically tight (in the batch size $K \to \infty$):

$$\lim_{K\to\infty} \mathcal{R}_{\theta,\nu_K^*}^{K,\text{con.}}(x, a) = \mathcal{R}_\theta^{\text{inv.}}(x, a) \tag{80}$$

Proof. Use Lemma 5 for the first part:

$$\mathcal{R}_{\theta,\nu}^{K,\text{con.}}(x, a) := \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \mathcal{L}_{\theta,\nu}^{K,\text{con.}}(x, a, Z) \tag{81}$$

$$= \mathbb{E}_{\substack{Z\sim p_\theta(\cdot|x,a) \\ Z_{1:K-1}\sim p_\theta^{K-1}}} \log \frac{e^{g_\nu(x,a,Z)}}{\frac{1}{K}\left(e^{g_\nu(x,a,Z)} + \sum_{i=1}^{K-1} e^{g_\nu(x,a,Z_i)}\right)} \tag{82}$$

$$\le \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \text{PMI}_\theta(x, a; Z) \tag{83}$$

$$=: \mathcal{R}_\theta^{\text{inv.}}(x, a) \tag{84}$$

and use Lemma 6 for the second part:

$$\lim_{K\to\infty} \mathcal{R}_{\theta,\nu_K^*}^{K,\text{con.}}(x, a) := \lim_{K\to\infty} \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \mathcal{L}_{\theta,\nu_K^*}^{K,\text{con.}}(x, a, Z) \tag{85}$$

$$= \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \lim_{K\to\infty} \mathcal{L}_{\theta,\nu_K^*}^{K,\text{con.}}(x, a, Z) \tag{86}$$

$$= \mathbb{E}_{Z\sim p_\theta(\cdot|x,a)} \text{PMI}_\theta(x, a; Z) \tag{87}$$

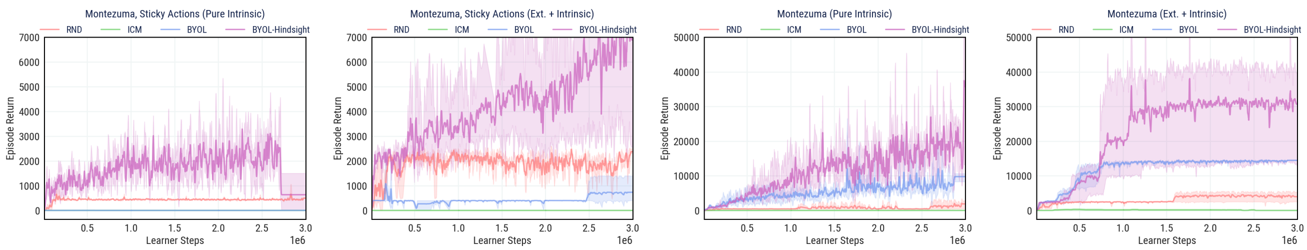
$$=: \mathcal{R}_\theta^{\text{inv.}}(x, a) \tag{88}$$

which completes the proof. □

B. Further Experiment Results

B.1. Sticky Actions

Figure 8 showed exploration as measured by the number of different dungeon rooms the agent manages to discover. For completeness, Figure 10 here also shows a comparison using the extrinsic reward that the agent obtains as a proxy. The conclusions are similar: In "Sticky Actions", `BYOL-Explore` instantly flatlines in the intrinsic-only regime, and only does marginally better in the mixed regime. In contrast, `BYOL-Hindsight` achieves much higher scores in both regimes. Moreover, in the "Non-Sticky Baseline", in both regimes `BYOL-Hindsight` actually manages to do even better than the original performance of `BYOL-Explore`.



(a) **Sticky Actions (Intrinsic Only)** (b) **Sticky Actions (Ext. + Intrinsic)** (c) **Non-Sticky Baseline (Intrinsic Only)** (d) **Non-Sticky Baseline (Ext. + Intrinsic)**

Figure 10. Montezuma's Revenge, with Sticky Actions and Non-Sticky Baseline. Performance measured by extrinsic rewards in an episode.

B.2. Persistive Noise

While sticky actions is the standard protocol for adding stochasticity, we further design an especially challenging form of stochasticity, as follows. First, recall that each observation O_t in our Atari environment is a grayscale 84×84-pixel image. In the "Persistive Noise" setting, observations are corrupted by an additive layer of 84×84-pixel noise that persists across time, with each pixel evolving randomly according to distributions that depend on the actions selected by the agent at each time step. Specifically, the value of each pixel $(i,j) \in \mathbb{N}^2_{<84}$ of the (final) observation $\tilde{O}_t$ is computed as follows:

$$\tilde{O}_t^{(i,j)} := O_t^{(i,j)} + U_t^{(i,j)} \mod 256 \tag{89}$$

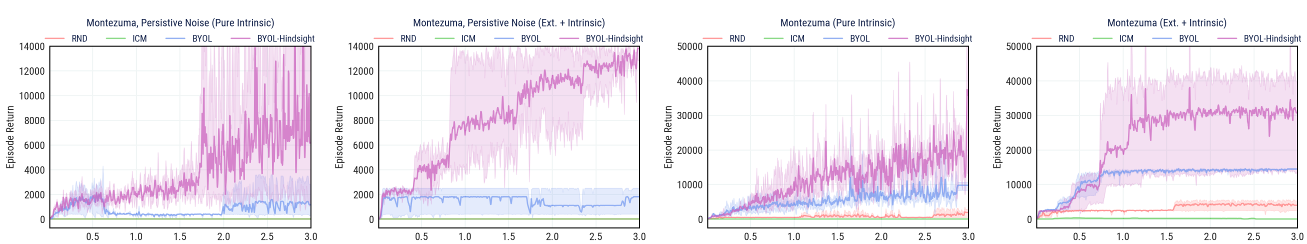
where U_t is the persistive layer of pixel noise:

$$U_t^{(i,j)} = U_{t-1}^{(i,j)} + \epsilon_t^{(i,j)} \mod 50 \tag{90}$$

and noise steps are sampled as:

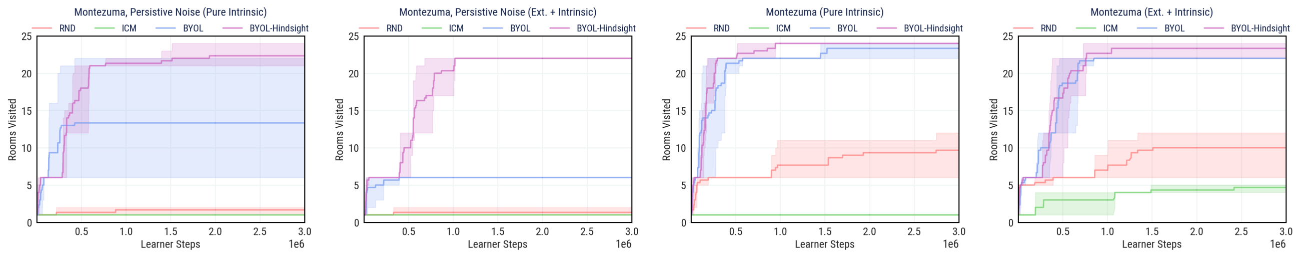
$$\epsilon_t^{(i,j)} \sim \text{Uniform}\{-1, 1\} \quad \text{if} \quad \texttt{key}(a_{t-1}) \text{ is odd} \tag{91}$$
$$\epsilon_t^{(i,j)} \sim \text{Uniform}\{-11, 11\} \quad \text{if} \quad \texttt{key}(a_{t-1}) \text{ is even} \tag{92}$$

where $\texttt{key}(a)$ is the numerical key code associated with action a. See Figure 5 for example frames generated by this process.



(a) **Persistive Noise (Intrinsic Only)** (b) **Persistive Noise (Ext. + Intrinsic)** (c) **Non-Noisy Baseline (Intrinsic Only)** (d) **Non-Noisy Baseline (Ext. + Intrinsic)**

Figure 11. Montezuma's Revenge, with Persistive Noise / Non-Noisy Baseline. Performance measured by extrinsic rewards in an episode.



(a) **Persistive Noise (Intrinsic Only)** (b) **Persistive Noise (Ext. + Intrinsic)** (c) **Non-Noisy Baseline (Intrinsic Only)** (d) **Non-Noisy Baseline (Ext. + Intrinsic)**

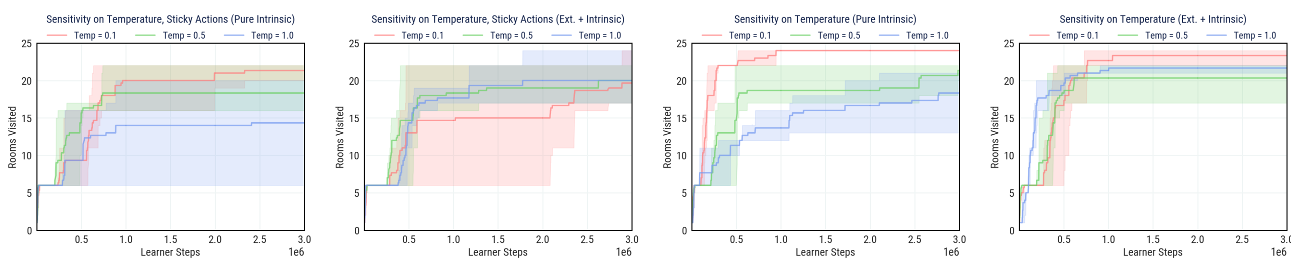
Figure 12. Montezuma's Revenge, with Persistive Noise / Non-Noisy Baseline. Performance measured by rooms reached (episodic setting).

Note that this setting is particularly challenging: While prior works have experimented with pixel-level noise (e.g. [11, 14]), they have either designed noise that is not additive (e.g. there are separate channels of pixels to the original frame) or not persistive (e.g. each time step's noise does not depend on the previous time step's noise)—which means it is in principle easy for a learned representation to simply "ignore" the noise. This is not possible in our setting.

Figures 11 and 12 show results (3M learner steps, 3 seeds), in terms of episode return as well as rooms visited. In addition to "Persistive Noise" as just described, the results for the vanilla environment are shown again for reference ("Non-Noisy Baseline"). The conclusion is as suspected: `BYOL-Explore` suffers greatly from the presence of this stochasticity, whereas `BYOL-Hindsight` is more resilient to it.

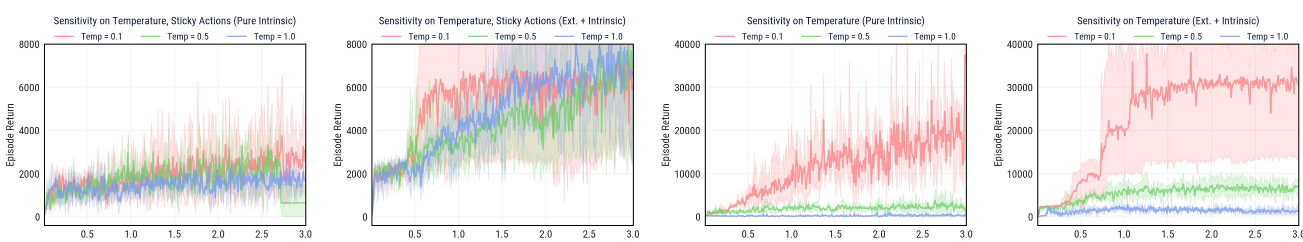
B.3. Temperature Sensitivity

The inner term in the contrastive loss (Objective 3) shares a similar form with contrastive losses in unsupervised representation learning [79–82], which admits a temperature (hyper-)parameter controlling the strength of penalties on negative samples [83]. A valid question is: How sensitive is `BYOL-Hindsight` to the specific choice of value for this parameter?



(a) **Sticky Actions (Intrinsic Only)** (b) **Sticky Actions (Ext. + Intrinsic)** (c) **Non-Sticky Baseline (Intrinsic Only)** (d) **Non-Sticky Baseline (Ext. + Intrinsic)**

Figure 13. Temperature Sensitivity (Montezuma's Revenge, with Sticky Actions and Non-Sticky Baseline). Rooms reached (episodic setting).



(a) **Sticky Actions (Intrinsic Only)** (b) **Sticky Actions (Ext. + Intrinsic)** (c) **Non-Sticky Baseline (Intrinsic Only)** (d) **Non-Sticky Baseline (Ext. + Intrinsic)**

Figure 14. Temperature Sensitivity (Montezuma's Revenge, with Sticky Actions and Non-Sticky Baseline). Extrinsic rewards in an episode.

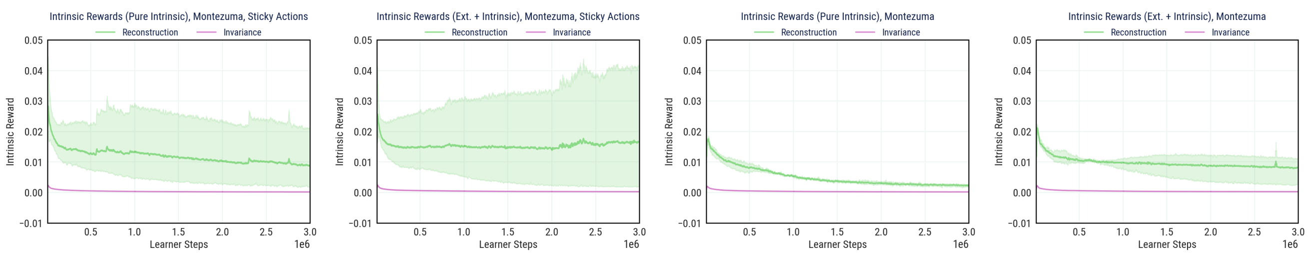
Figures 14 and 13 show results (3M learner steps, 3 seeds), in terms of episode return as well as rooms visited, for "Sticky Actions" and "Non-Sticky Baseline". Interestingly, the performance of `BYOL-Hindsight` in the "Sticky Actions" setting is only mildly sensitive to the choice of temperature. In the "Non-Sticky Baseline" setting, sensitivity is most acute in the intrinsic-only exploration regime: Lower temperature performs better than higher temperature. This makes sense, because in a largely deterministic environment, a lower temperature provides a stronger incentive for invariances to be enforced, whereas a higher temperature may allow more leakage of information from X, A into Z, which may diminish exploration.

B.4. Intrinsic Rewards

The derivation of Theorem 8 makes it clear that "hindsight information" and "total stochasticity" can be misaligned in two ways: First, hindsight may capture less information than in the "true" latent (which means the reconstruction error is not driven to zero). Second, hindsight may capture more information than in the "true" latent (which means it leaks some information about the current state and action).

Regarding the former, Figure 15 shows that—for both noise settings and both exploration regimes—the reconstruction bonus indeed converges to very small values, but not exactly zero, which is consistent with the fact that Z does not perfectly capture the entirety of what is unpredictable. (This may be due to a variety of usual factors, such as non-realizability and errors in optimization and estimation of expectations).

Regarding the latter, there are several ways to assess how well the invariance constraint between Z and X, A may be enforced. Firstly, we may observe the value of the invariance loss. Figure 15 also shows the invariance bonus over time: We observe that—to the best of the critic's ability to tell—the invariance constraint appears relatively well-enforced. Of course, this is certainly not a perfect measure, as it depends on how discriminative the critic is, to begin with.

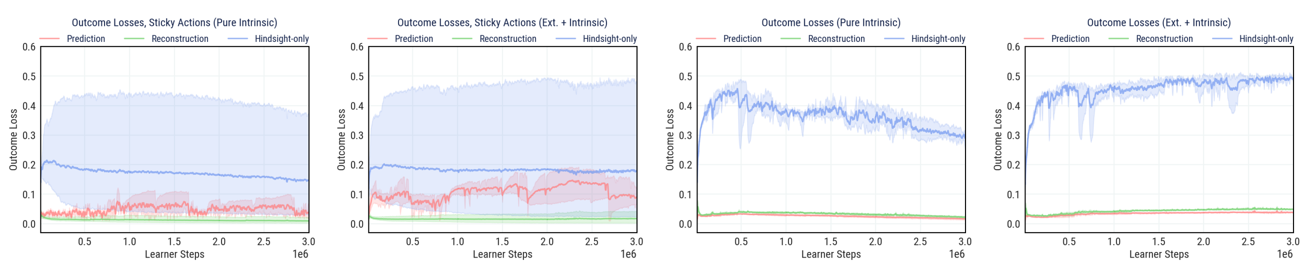


(a) **Sticky Actions (Intrinsic Only)** (b) **Sticky Actions (Ext. + Intrinsic)** (c) **Non-Sticky Baseline (Intrinsic Only)** (d) **Non-Sticky Baseline (Ext. + Intrinsic)**

Figure 15. Intrinsic Rewards (Montezuma's Revenge, with Sticky Actions and Non-Sticky Baseline). Reconstruction and invariance bonuses.

B.5. Outcome Losses

Secondly, we can also gauge the amount of informational overlap between Z and X, A as follows: In addition to learning the function $f_\eta(X, A, Z)$ (viz. "Reconstruction Loss"), consider training additional predictors to predict Y, but only using states and actions as input as in the usual forward prediction (viz. "Prediction Loss"), as well as only using the learned hindsight vectors as input (viz. "Hindsight-only Loss").



(a) **Sticky Actions (Intrinsic Only)** (b) **Sticky Actions (Ext. + Intrinsic)** (c) **Non-Sticky Baseline (Intrinsic Only)** (d) **Non-Sticky Baseline (Ext. + Intrinsic)**

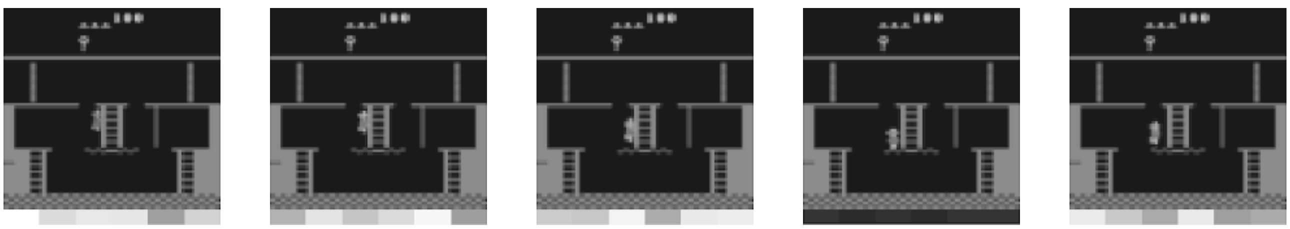
Figure 16. Prediction, Reconstruction, and Hindsight-only Losses (Montezuma's Revenge, with Sticky Actions and Non-Sticky Baseline).

From Figure 16, we observe in the "Sticky Actions" setting that the hindsight-only error (i.e. using only Z as input) is the highest; prediction error (i.e. using only X, A) is lower, but not as low as reconstruction error (i.e. using X, A, Z as input). In the "Non-Sticky Baseline" setting, as expected the prediction and reconstruction errors are roughly equal, since the environment is largely deterministic, so adding whatever hindsight vectors as input will not confer any benefit in modeling outcomes. For reference, the variance of target vectors is around 0.4. These observations are consistent with the fact that leakage indeed occurs, but much of it is regularized away by the invariance constraint. Precisely, from the loss comparison we see that Z alone contains strictly less information than in X, A for predicting Y, and that their union X, A, Z contains the most information for predicting Y. (Note that much information about each room is statically determined by the room number, thus even a tiny amount if leakage may be sufficient to determine large portions of the future).

Finally, we stress that care is required when interpreting these curves due to the bootstrapped nature of the latent space in `BYOL-Hindsight`, as well as the fact that the dataset on which the predictor/reconstructor, generator, and critic are trained is endogenous to the rollout policy trained on the basis of the intrinsic reward.

B.6. Hindsight Information

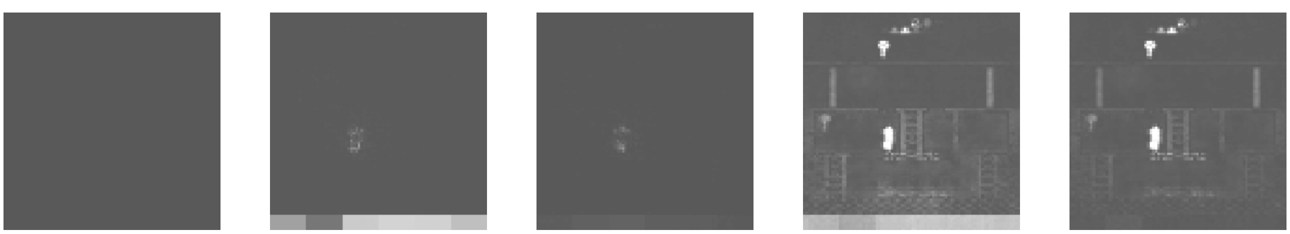
Thirdly, we can visually inspect for informational overlap in a simulation setting, as follows: In a rollout dataset ρ_π, define input states as the most recent four frames, $x_t := o_{t-3:t}$, and define target states as the next frame $x_{t+1} := o_{t+1}$. Then we learn representations z_{t+1} as usual—that is, by optimizing the objective $\min_{\theta,\eta}\max_\nu \mathbb{E}_{X_t, A_t \sim \pi(\cdot|X_t)} \mathcal{R}^K_{\theta,\eta,\nu}(X_t, A_t)$. As our source of stochasticity, we use a strip of large "patches" at the bottom of each frame, whose grayscale values are random variables that depend on the action selected by the agent in the prior time step. Importantly, each action may induce a different distribution of patch values for the strip that appears in the next frame—that is, the most natural parameterization involves a structural causal model with directed edges from A_t to Z_{t+1}. See Figure 17 for a representative example of such a sequence of observations. (Note that this artificial-noise setting is similar to the noisy pixels we used above in Appendix B.2, but these noise patches are larger and so more discernible than noise pixels for visual inspection).



(a) $o_{t-3}; a_{t-4} = R$ (b) $o_{t-2}; a_{t-3} = R$ (c) $o_{t-1}; a_{t-2} = R$ (d) $o_t; a_{t-1} = L$ (e) $o_{t+1}; a_t = U$

Figure 17. Example Input State ($x_t := o_{t-3:t}$) and Target State ($x_{t+1} := o_{t+1}$). For each frame, the prior (patch-sampling) action is shown.

Then we learn the following five functions: (a) "Identity", i.e. $\hat{x}_{t+1} := h(x_{t+1})$; (b) "Prediction", i.e. $\hat{x}_{t+1} := h(x_t, a_t)$; (c) "Reconstruction", i.e. $\hat{x}_{t+1} := h(x_t, a_t, z_{t+1})$; (d) "Hindsight-Only", i.e. $\hat{x}_{t+1} := h(z_{t+1})$; and (e) "Hindsight-and-Action-Only", i.e. $\hat{x}_{t+1} := h(a_t, z_{t+1})$. Given a new input state x_t and action a_t (and hindsight z_{t+1} from the generator), we obtain the outputs $\hat{x}_{t+1}$ from these functions, and inspect the pixel-wise differences (i.e. target state errors) from ground-truths x_{t+1}.



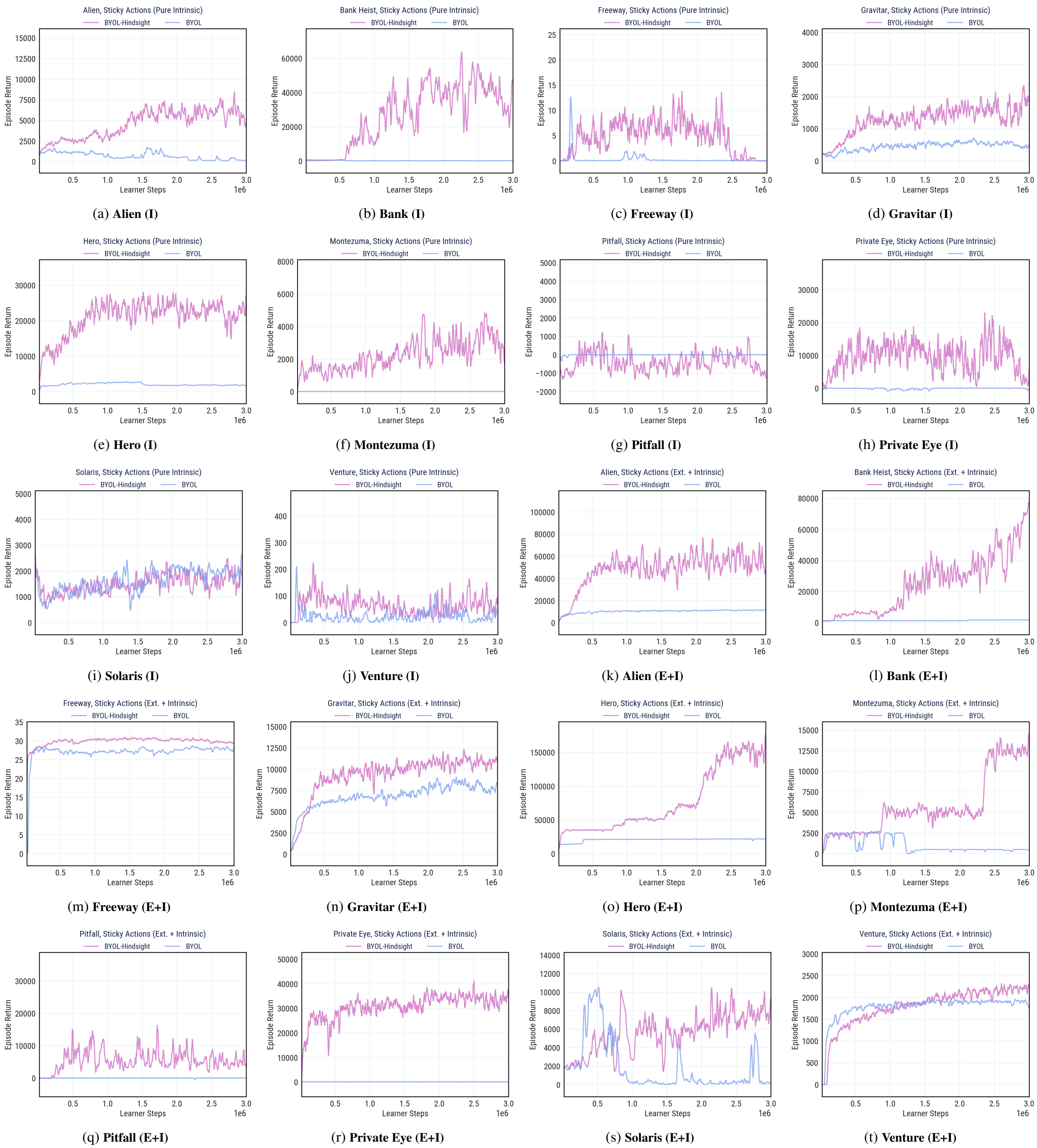
(a) $\hat{x}_{t+1} := h(x_{t+1})$ (b) $\hat{x}_{t+1} := h(x_t, a_t)$ (c) $\hat{x}_{t+1} := h(x_t, a_t, z_{t+1})$ (d) $\hat{x}_{t+1} := h(z_{t+1})$ (e) $\hat{x}_{t+1} := h(a_t, z_{t+1})$

Figure 18. Visualizing Target State Errors. Identity, Prediction, Reconstruction, Hindsight-Only, and Hindsight-and-Action-Only Errors.

See Figure 18 for a representative example (corresponding to the example in Figure 17). Note that the first three results are straightforward and as expected: (a) The "Identity" function appears to be learned very well. (b) The "Prediction" function also appears to learn to predict the main content of the next frame quite well, but is completely unable to predict the random pixel strip at the bottom of the frame. (c) The "Reconstruction" function appears to learn to reconstruct both the main content of the next frame, as well as the random pixel strip. Importantly, we may now ask: What if z_{t+1} has simply learned to copy x_{t+1}, or otherwise leaked information from x_t, a_t? The next two results reassure us that this is unlikely: (d) The "Hindsight-Only" function appears to map quite poorly into both the main content and the pixel strip (e.g. the main character is never even there), which tells us that z_{t+1} alone does not have great overlap with x_t, a_t. Moreover, (e) The "Hindsight-and-Action-Only" function appears to map quite poorly into the main content, but now the pixel strip is actually modeled very well, which is consistent with the fact that z_{t+1} does capture latent information about stochasticity that relies on a_t for resolution.

B.7. Hard Exploration Games

Figure 9 only showed results for Atari hard-exploration games in the intrinsic-only "(I)" exploration regime, due to space constraints. Figure 19 shows larger plots for those, as well as corresponding results for the mixed "(E+I)" exploration regime.



(a) **Alien (I)** (b) **Bank (I)** (c) **Freeway (I)** (d) **Gravitar (I)**

(e) **Hero (I)** (f) **Montezuma (I)** (g) **Pitfall (I)** (h) **Private Eye (I)**

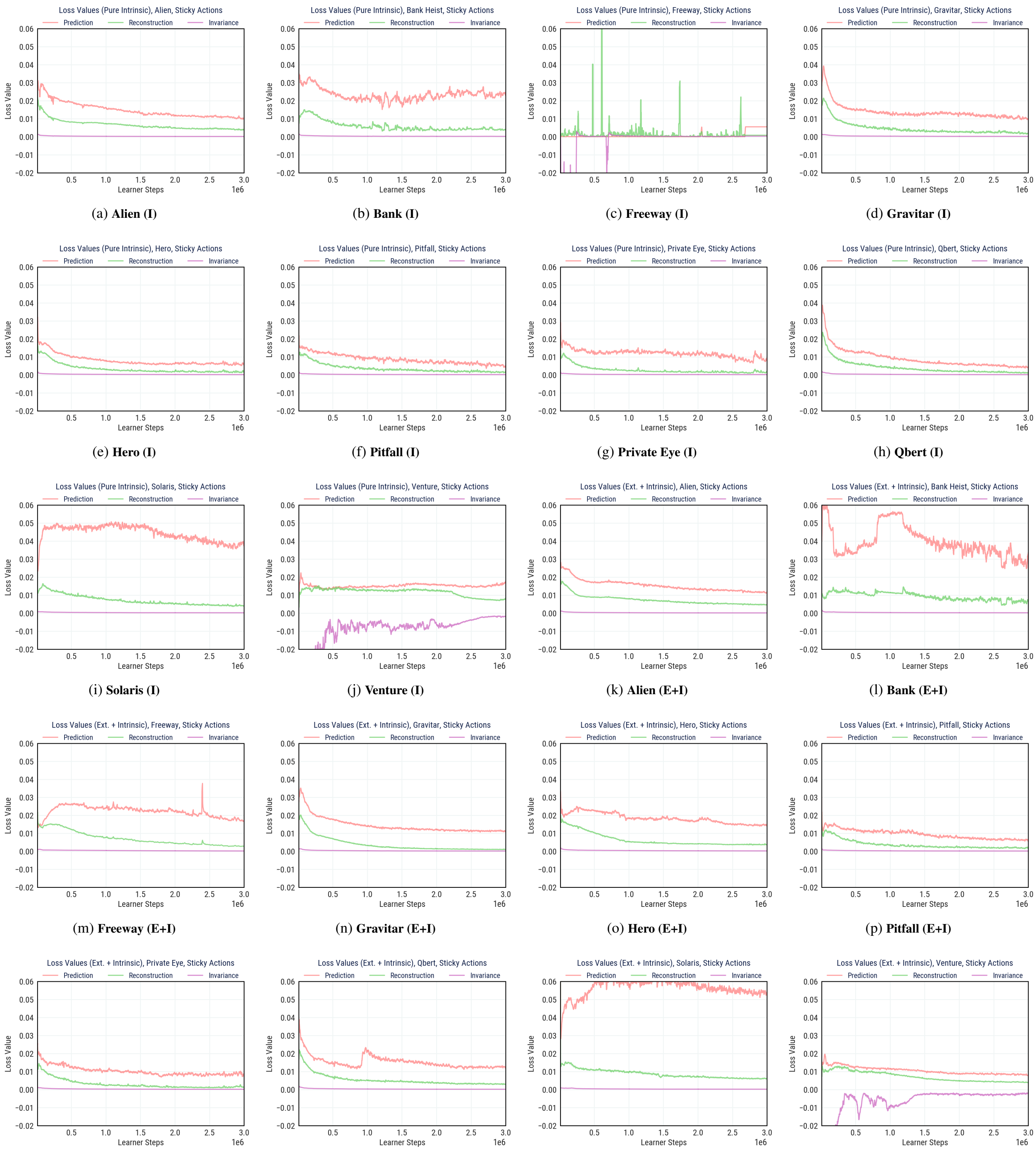
(i) **Solaris (I)** (j) **Venture (I)** (k) **Alien (E+I)** (l) **Bank (E+I)**

(m) **Freeway (E+I)** (n) **Gravitar (E+I)** (o) **Hero (E+I)** (p) **Montezuma (E+I)**

(q) **Pitfall (E+I)** (r) **Private Eye (E+I)** (s) **Solaris (E+I)** (t) **Venture (E+I)**

Figure 19. Hard Exploration Games, with Sticky Actions. Performance measured by the sum of extrinsic rewards obtained in an episode.

For additional insight into loss dynamics, Figure 20 shows the behavior of `BYOL-Hindsight`'s reconstruction and invariance losses. For comparison, a predictor is also trained to measure the usual forward prediction loss. We observe that the losses generally behave consistently with our hypothesis: Prediction losses are higher than reconstruction losses due to stochasticity.



(a) **Alien (I)** (b) **Bank (I)** (c) **Freeway (I)** (d) **Gravitar (I)**

(e) **Hero (I)** (f) **Pitfall (I)** (g) **Private Eye (I)** (h) **Qbert (I)**

(i) **Solaris (I)** (j) **Venture (I)** (k) **Alien (E+I)** (l) **Bank (E+I)**

(m) **Freeway (E+I)** (n) **Gravitar (E+I)** (o) **Hero (E+I)** (p) **Pitfall (E+I)**

(q) **Private Eye (E+I)** (r) **Qbert (E+I)** (s) **Solaris (E+I)** (t) **Venture (E+I)**

Figure 20. Loss Values (Hard Exploration Games, with Sticky Actions). `BYOL-Hindsight` prediction, reconstruction, invariance losses.

B.8. Additional Remarks on Results

Bank Heist *In Figure 7, why are the returns higher for natural traps (intrinsic only) when compared with natural traps (intrinsic + extrinsic)?*

First, note in Figure 7(a)–(b) that for both the "intrinsic-only" and "mixed" regimes, the agent is still improving over time, except that the latter is improving more slowly. This phenomenon is actually not necessarily surprising.

Consider training an agent in the "intrinsic-only" and "mixed" regimes. Indeed, as training proceeds over time, we may generally expect that returns obtained in the latter eventually surpass returns obtained in the former—because it has access to the extrinsic reward signal itself. However, the key word here is "*eventually*": In general, it is not always true that an agent in the "mixed" regime improves *more quickly* than in the "intrinsic-only" regime. How quickly their returns improve with/without extrinsic rewards depends entirely on the specifics of the environment, including how the extrinsic rewards are distributed, as well as how their magnitudes compare to the intrinsic rewards from exploration. For example, if there are many opportunities for earning small extrinsic rewards, then in the "mixed" regime the agent may spend more time chasing after those extrinsic rewards during training, which may slow down their exploration of further parts of the environment that could actually yield more rewards later on.

The Bank Heist environment is characterized by this, which offers a plausible explanation for the observed result: The game consists of a series of different cities that can be entered and exited in sequence. In each city, extrinsic rewards are obtained by robbing banks (by running over them) and also by destroying police cars (by dropping dynamite onto them). In each city, police cars respawn over time, and banks respawn when police cars are destroyed, so there are many opportunities for small extrinsic rewards to be earned, unless the agent is caught or runs out of fuel. Now, in the "intrinsic-only" regime, we expect that the agent is rewarded greatly for entering each new city (which loads an entirely different maze onto the screen). In order to keep exploring new cities, the agent must learn to *survive*, which requires learning to destroy police cars and rob banks to refuel on exit. So, pure exploration in Bank Heist is already aligned with episode returns. On the other hand, in the "mixed" regime, the agent is *additionally* incentivized to maximize the number of banks robbed and police cars destroyed in any given city, because the number of police cars and banks robbed nonlinearly compound the extrinsic rewards earned in that city. However, doing so increases the risk of being caught or running out of fuel, which may slow down learning.

In Figure 7(a)–(b), we see that the "mixed" agent starts earning non-trivial returns earlier than the "intrinsic-only" agent: At 1.2m learner steps, almost 20,000 is earned by the former, compared to 2,000 by the latter. Subsequently, the former improves more slowly than the latter. But given the above discussion, this result is consistent with the observation that the "mixed" agent spends more time within each city trying to destroy more police cars and rob more banks, more so than the "intrinsic-only" agent, which is only incentivized to do so "sufficiently" for surviving to explore more cities—thereby incidentally earning more rewards in further cities as a side-effect. Again, note that we do expect the "mixed" agent to eventually surpass the "intrinsic-only" agent, but this may take much longer to happen (i.e. beyond the 3m learner steps in the experiment). Of course, the precise dynamics of learning depends on the coefficient that combines the intrinsic and extrinsic rewards. In this work, we do not tune this mixing coefficient for each individual environment for optimal learning speed, because our focus is instead simply on demonstrating a positive benefit of *hindsight* for curiosity-driven exploration.

(Note that this phenomenon is not observed for the Sticky Actions setting, which makes sense: Combined with the fact that dynamite explodes at unpredictable times, the fact that actions are sticky means that precise timing in destroying police cars is impossible in this setting, moreover exiting the maze to refuel and visit a new city may require several drive-bys to succeed, therefore staying too long in a city becomes very dangerous. So in this case, the policy has extrinsic incentive to exit and refuel to new cities earlier than before, which counteracts the effect described above. This gives a plausible explanation for the observation that in this setting, the "mixed" agent spends less time than before in each city, but explores more cities and ends up earning more returns more quickly).

Montezuma's Revenge *In Figures 8 and 10, for sticky actions, why is the number of rooms visited reduced by around 25%, whereas the returns are reduced by around 75%?*

Note that Figure 8 measures the number of different rooms the agent manages to discover over the training run, whereas Figure 10 measures the mean episodic return that the agent obtains. (This is similar to prior work, such as in [12] and [14]). Therefore they are not necessarily one-to-one proportional to each another, as the agent may successfully find more and more later rooms over training, while still spending the majority of its time in earlier rooms.

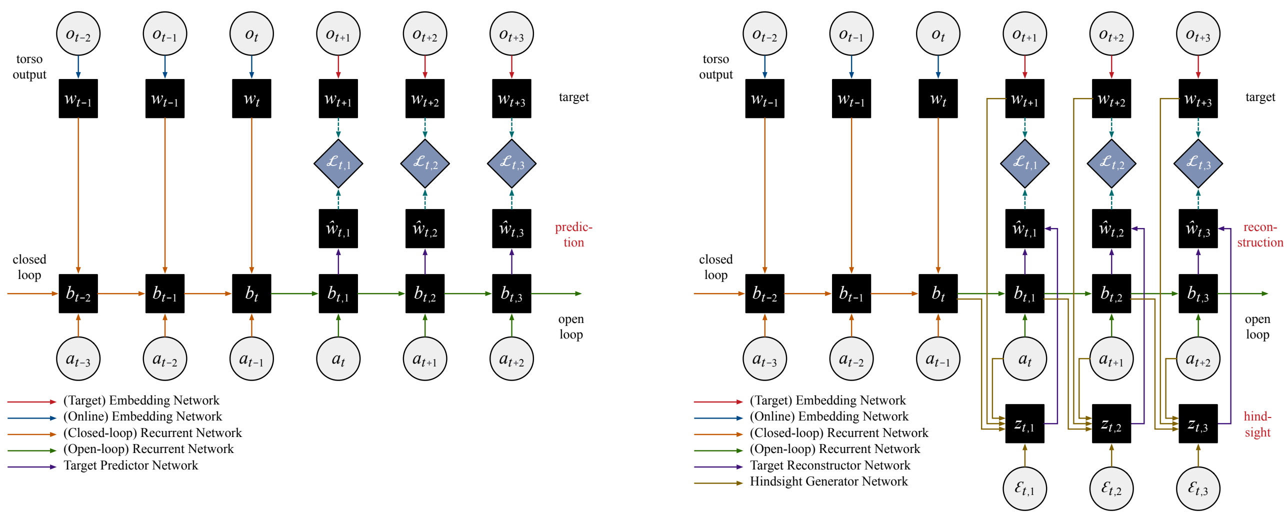
Broadly speaking, sticky actions have two negative effects on gameplay. First, it obviously adds *randomness* to the environment's dynamics, which throws off curiosity-driven exploration (e.g. BYOL-Explore) since sticky actions are a source of stochastic traps. Second, it also simply makes playing the game *more difficult*, since the agent has less control over what actions are actually executed (e.g. they can die more easily due to unfortunate sticky actions). Now, what BYOL-Hindsight does is mitigate the first problem, such that the agent's policy is optimized using intrinsic rewards that are more or less unaffected by action stickiness. However, it cannot change the fact that the game is in fact more difficult to play, which means that progressing/staying alive is harder. So while the agent may still manage to discover many rooms, it may spend most of its time in earlier rooms, so the mean episodic return is lower.

Consider a "heatmap" of the rooms visited in the game over the training run. Compared to the non-sticky setting, in the sticky setting the heatmap has higher heat in the earlier rooms, and lower heat in the later rooms. However, the number of rooms with non-zero heat do not differ by much.

Random Network Distillation *In Figure 6, why does the performance of RND drop suddenly after ∼400K steps?*

This is simply the "vanishing rewards" phenomenon (see e.g. [34] for this terminology). Many methods relying on some notion of "novelty" for intrinsic rewards tend to exhibit this over time: After the novelty of a state has vanished, the agent is not incentivized to visit it again. In a small environment—such as the Pycolab Maze environment in Figure 6—if rewards vanish for all states, then the agent is not incentivized to do anything at all. Note that RND is especially susceptible to this, since (unlike the other algorithms) it does not need to learn any dynamics mapping from histories to future states, but rather it is simply learning a mapping from x_{t+1} to $f_{\text{random}}(x_{t+1})$ for some initially unknown f_{random}. So while its errors *initially* provide good incentive to explore, it much more quickly "loses interest" due to vanished rewards, hence the observed drops.

C. Further Implementation Detail



(a) **BYOL-Explore** (b) **BYOL-Hindsight**

Figure 21. Neural Architecture of BYOL-Explore and BYOL-Hindsight.

In all experiments, start from the same architecture/hyperparameters for `BYOL-Explore` as specified in [14], including target network EMA, open-loop horizon, intrinsic reward normalization/prioritization, representation sharing, and underlying RL algorithm. In the following, we first recall the architecture of `BYOL-Explore` in detail, then describe `BYOL-Hindsight`:

C.1. BYOL-Explore

Online Embedding Network. Figure 21(a) shows the architecture of `BYOL-Explore`. First, to compute predictions of target states, an online network is composed of an encoder ω that transforms observations o_t into representations $w_t = \omega(o_t)$.

Closed-Loop Recurrent Network. Next, a closed-loop RNN computes representations b_t on the basis of previous actions $\{a_{t'}\}_{t'<t}$ and observation encodings $\{w_{t'}\}_{t'\leq t}$. Specifically, the closed-loop RNN cell takes in each observation representation w_t, previous action a_{t-1}, and previous belief b_{t-1} as input, and computes a representation b_t of the history so far.

Open-Loop Recurrent Network. Then, an open-loop RNN is initialized by this b_t, and computes forward predictions $b_{t,i}$ for horizon steps indexed as i, on the basis of actions $\{a_{t'}\}_{t'\geq t}$ up to some maximum open-loop horizon. Specifically, the open-loop RNN cell takes in each action a_{t+i-1} and current belief $b_{t,i-1}$ as input, and computes a representation $b_{t,i}$ of the predicted next belief. The purpose of this is to simulate future beliefs using only knowledge of future actions.

Target Predictor Network. Lastly, a predictor network takes the open-loop belief $b_{t,i}$ as input, and outputs the open-loop (raw) prediction $\hat{w}_{t,i}$. (We say "raw" here in order to distinguish from the normalized predictions $\hat{x}_{t,i}$ below).

Target Embedding Network. Corresponding to the online network is a target network whose parameters are an exponential moving average of the parameters of the online network. The target network encodes observations o_{t+i} as $w_{t+i} = \omega_{\text{target}}(o_{t+i})$, and these targets are used to train the online network. The weights of ω_{target} are updated per the averaging rule $\omega_{\text{target}} \leftarrow \alpha\omega_{\text{target}} + (1-\alpha)\omega$ after each training step, with α being the exponential moving average parameter.

Loss Function. Define target states as the ℓ_2-normalized encodings $x_{t+i} \coloneqq \texttt{sg}(w_{t+i}/\|w_{t+i}\|_2)$ of future observations, and predictions as the ℓ_2-normalized (raw) predictions $\hat{x}_{t,i} \coloneqq \hat{w}_{t,i}/\|\hat{w}_{t,i}\|_2$. Moreover, define input states as the beliefs themselves—that is, $x_{t,i-1} \coloneqq b_{t,i-1}$. Then, the loss function used to train all networks (except the target network) is given by $\mathcal{R}(x^{(j)}_{t,i-1}, a^{(j)}_{t+i-1}) \coloneqq \|x^{(j)}_{t+i} - \hat{x}^{(j)}_{t,i}\|_2^2$ where j indexes the trajectories within a batch. The overall objective for training the networks is the average over the time, batch, and horizon dimensions.

Intrinsic Reward. Finally, the intrinsic reward associated to each observed transition $(o^{(j)}_s, a^{(j)}_s, o^{(j)}_{s+1})$ is the sum of corresponding prediction errors $\sum_{t+i=s+1}\mathcal{R}(x^{(j)}_{t,i-1}, a^{(j)}_{t+i-1})$, which aggregates all the errors pertaining to the world model relative to the observation $o^{(j)}_{s+1}$—the intuition being that the intrinsic reward for a time step is proportional to how difficult it is to predict its observation from partial histories. See Algorithm 1.

C.2. BYOL-Hindsight

Target Reconstructor Network. Figure 21(b) shows the architecture of `BYOL-Hindsight`. Starting from the setup for `BYOL-Explore`, the modification to incorporate hindsight is as follows: First, the predictor network is now replaced by a reconstructor network, which takes the open-loop belief $b_{t,i}$ and hindsight vector $z_{t,i}$ as input, and outputs the open-loop (raw) reconstruction $\hat{w}_{t,i}$. (Notation: At this point, it is helpful to recall that the online embedding network, closed-loop RNN, open-loop RNN, and target predictor/reconstructor network all correspond to what we subsume under parameter η).

Generator and Critic Networks. Second, a hindsight generator network p_θ takes in the belief $b_{t,i-1}$, the action a_{t+i-1}, and the target w_{t+i}, and samples a hindsight vector $Z_{t,i} \sim p_\theta(\cdot|b_{t,i-1}, a_{t+i-1}, w_{t+i})$ by taking an additional noise vector $\varepsilon_{t,i}$ as input. Finally, a hindsight critic network g_ν takes in any belief $b_{t,i-1}$, any action a_{t+i-1}, and any hindsight vector $z_{t,i}$ as input, and outputs the corresponding energy $g_\nu(b_{t,i-1}, a_{t+i-1}, z_{t,i})$.

Loss Function. There are now two components. Firstly, analogous to before, define target states as the ℓ_2-normalized encodings $x_{t+i} \coloneqq \texttt{sg}(w_{t+i}/\|w_{t+i}\|_2)$ of future observations, and reconstructions as the ℓ_2-normalized (raw) reconstructions $\hat{x}_{t,i} \coloneqq \hat{w}_{t,i}/\|\hat{w}_{t,i}\|_2$. Moreover, define input states as the beliefs themselves—that is, $x_{t,i-1} \coloneqq b_{t,i-1}$. Then, the loss function used to train the online embedding network, closed-loop RNN, open-loop RNN, target reconstructor network, and hindsight generator network (i.e. all networks except the target network and critic network) is given by: $\mathcal{R}^{\text{rec.}}_{\theta,\eta}(x^{(j)}_{t,i-1}, a^{(j)}_{t+i-1}) \coloneqq \|x^{(j)}_{t+i} - \hat{x}^{(j)}_{t,i}\|_2^2$ where j indexes the trajectories within a batch. The overall (reconstructive) objective for training the networks is the average over the time, batch, and horizon dimensions. Second, the critic needs to ensure that $Z_{t,i}$ be independent of $X_{t,i-1}, A_{t+i-1}$. The loss function used to train the hindsight generator and critic networks is given by: $\mathcal{R}^{K,\text{con.}}_{\theta,\nu}(x^{(j)}_{t,i-1}, a^{(j)}_{t+i-1}) \coloneqq \log\big[e^{g_\nu(x^{(j)}_{t,i-1}, a^{(j)}_{t+i-1}, z^{(j)}_{t,i})} \big/ \tfrac{1}{K}\big(e^{g_\nu(x^{(j)}_{t,i-1}, a^{(j)}_{t+i-1}, z^{(j)}_{t,i})} + \sum_{k=1}^{K-1} e^{g_\nu(x^{(j)}_{t,i-1}, a^{(j)}_{t+i-1}, z^{(k)}_{t,i})}\big)\big]$, where k is another index into the trajectories within a batch. The overall (contrastive) objective for training the networks is the average over the time, batch, and horizon dimensions.

Intrinsic Reward. Finally, analogous to before, the intrinsic reward associated to each observed transition $(o^{(j)}_s, a^{(j)}_s, o^{(j)}_{s+1})$ is the sum of corresponding reconstruction+contrastive errors $\sum_{t+i=s+1}\big[\frac{1}{\lambda}\mathcal{R}^{\text{rec.}}_{\theta,\eta}(x^{(j)}_{t,i-1}, a^{(j)}_{t+i-1}) + \mathcal{R}^{K,\text{con.}}_{\theta,\nu}(x^{(j)}_{t,i-1}, a^{(j)}_{t+i-1})\big]$, which aggregates all the errors pertaining to the world model relative to the observation $o^{(j)}_{s+1}$—the intuition being that the intrinsic reward for a time step is proportional to how difficult it is to reconstruct its observation from partial histories as well as how difficult it is to generate hindsight representations disentangled from input states and actions. See Algorithm 2.

C.3. RL Hyperparameters

Like with `BYOL-Explore`, any RL algorithm can be used in conjunction with `BYOL-Hindsight`. We use VMPO [75] exactly as specified in [14], and reproduce the details as follows: PopArt-style [96] reward normalization is used with step size 0.01, and rewards are subsequently rescaled by $1-\gamma$ with discount factor $\gamma = 0.999$. PopArt normalization is also applied to the output of the value network. To train the value function, VTrace is used without off-policy correction to define temporal-difference targets for mean squared error loss with loss weight 0.5, and an entropy loss with loss weight 0.001 is added. The parameters η_{init} and α_{init} for VMPO are initialized to 0.5, and $\epsilon_\eta = 0.01$ and $\epsilon_\alpha = 0.005$. The top-k parameter for VMPO is set to 0.5. For optimization, the Adam optimizer is used with learning rate 10^{-4} and $b_1 = 0.9$.

Algorithm 1 BYOL-Explore

```
repeat                                          ▷ batch indices j suppressed unless explicitly required
  execute policy π to obtain dataset of actions a and observations o
  for j = 1, ... do                                                                   ▷ batch index
    compute online embedding, ω_t ← ω(o_t) for all time steps t
    for t = 1, ... do                                                                 ▷ time step
      compute closed-loop belief, b_t ← RNN_closed-loop({a_t'}_{t'<t}, {ω_t'}_{t'≤t})
      for i = 1, ... do                                                                 ▷ horizon
        compute open-loop belief, b_{t,i} ← RNN_open-loop(b_t, {a_t'}_{i>t'≥t})
        compute target embedding, ω_{t+i} ← ω_target(o_{t+i}) and normalized target, x_{t+i} ← sg(w_{t+i}/‖w_{t+i}‖_2)
        compute predicted embedding, ŵ_{t,i} ← ψ(b_{t,i}) and normalized prediction, x̂_{t,i} := ŵ_{t,i}/‖ŵ_{t,i}‖_2
      end for
    end for
  end for
  update ω, RNN_closed-loop, RNN_open-loop, ψ using R(x^(j)_{t,i−1}, a^(j)_{t+i−1}) := ‖x^(j)_{t+i} − x̂^(j)_{t,i}‖²_2 averaged over i, j, t
  update ω_target using exponential moving averaging, ω_target ← αω_target + (1 − α)ω
  update π using R_intrinsic(o_s, a_s, o_{s+1}) := Σ_{t+i=s+1} R(x_{t,i−1}, a_{t+i−1})
end repeat
```

The VMPO target network is updated every 10 learner steps. In terms of computation, 400 CPU actors generate data through an inference server, using four TPUv2 for evaluating the policy. Curiosity in Hindsight is agnostic to the underlying reinforcement learning algorithm used to optimize intrinsic rewards, so all RL implementation details in `BYOL-Hindsight` are identical to those in the original `BYOL-Explore` experiments.

C.4. BYOL Hyperparameters

For the components of `BYOL-Hindsight` that overlap with `BYOL-Explore`, we use the exact same architecture and hyperparameters. Observation representations of size 512 and history representations of size 256. The encoder is a Deep ResNet stack [97], with grayscale image observations passing through a stack of 3 units, each made up of a 3×3 convolutional layer, a 3×3 max pool layer, and two residual blocks. The convolutional layer and residual blocks have number of channels (16, 32, 32) for each of the 3 units. GroupNorm normalization [98] is used with one group at the end of each unit, and ReLU activations are used everywhere. At the end of the final residual block, the output is flattened and projected with a single linear layer to embedding dimension 512. The closed-loop and open-loop RNNs are simple GRUs [99], with actions provided to the RNN cells, embedded to representation size 32. The policy head and value head are MLPs with one hidden layer of size 256, and the outputs of the policy head are passed through a softmax layer to obtain action probabilities. Specifically for `BYOL-Explore` only, the predictor network is an MLP with three hidden layers of size 512 (Note that this detail is the first of two that are different to the original implementation in [14]; using three layers of 512 instead of the original single layer of 256 leads to better results). In the mixed exploration regime, the intrinsic/extrinsic rewards mixing coefficient is 0.2 (This is the second of two details that are different to the original implementation in [14]; using 0.2 instead of the original 0.1 leads to better results). We defer to [14] for details on intrinsic reward normalization/prioritization and representation sharing. We use the classical 30 random no-ops evaluation regime for Atari [100, 101]. The batch size is 32 and sequence length is 128, and four TPUv2 are used in a distributed learning setup. The open loop horizon is 1 for all Pycolab experiments, and 8 for all Atari experiments. The target network EMA is 0.99.

C.5. Hindsight Hyperparameters

Specifically for `BYOL-Hindsight`, the reconstructor network is an MLP with three hidden layers of 512, which is the same as the predictor network in `BYOL-Explore` above. The generator network and critic network are MLPs with three hidden layers of 512. The dimension of the generator noise ϵ is 256, and the dimension of the hindsight vector is 256. The temperature parameter is 0.5, except in Montezuma's revenge where we show sensitivity to the temperature. The coefficient $\lambda = 1$ for model learning. For policy optimization, we empirically observe that the value of λ has little to no contribution towards the intrinsic reward (and little to no effect on exploration); for simplicity we set λ to zero for policy optimization. For the contrastive loss, negative samples are simply taken from the batch, so the contrastive set is also batch size 32; the

Algorithm 2 BYOL-Hindsight

```
repeat                                                    ▷ batch indices j suppressed unless explicitly required
  execute policy π to obtain dataset of actions a and observations o
  for j = 1, ... do                                                                            ▷ batch index
    compute online embedding, ω_t ← ω(o_t) for all time steps t
    for t = 1, ... do                                                                           ▷ time step
      compute closed-loop belief, b_t ← RNN_closed-loop({a_t'}_{t'<t}, {ω_t'}_{t'≤t})
      for i = 1, ... do                                                                          ▷ horizon
        compute open-loop belief, b_{t,i} ← RNN_open-loop(b_t, {a_t'}_{i>t'≥t})
        compute target embedding, ω_{t+i} ← ω_target(o_{t+i}) and normalized target, x_{t+i} ← sg(w_{t+i}/‖w_{t+i}‖_2)
        sample hindsight vector, Z_{t,i} ~ p_θ(·|b_{t,i-1}, a_{t+i-1}, w_{t+i})
        evaluate hindsight energy, g_ν(b_{t,i-1}, a_{t+i-1}, z_{t,i})
        compute reconstructed embedding, ŵ_{t,i} ← ψ(b_{t,i}, z_{t,i}) and normalized prediction, x̂_{t,i} := ŵ_{t,i}/‖ŵ_{t,i}‖_2
      end for
    end for
  end for
  update ω, RNN_closed-loop, RNN_open-loop, ψ, p_θ using R^rec._{θ,η}(x^(j)_{t,i-1}, a^(j)_{t+i-1}) := ‖x^(j)_{t+i} − x̂^(j)_{t,i}‖_2^2 averaged over i, j, t
  update p_θ, g_ν using R^{K,con.}_{θ,ν}(x^(j)_{t,i-1}, a^(j)_{t+i-1}) := log [e^{g_ν(x^(j)_{t,i-1}, a^(j)_{t+i-1}, z^(j)_{t,i})} / (1/K)(e^{g_ν(x^(j)_{t,i-1}, a^(j)_{t+i-1}, z^(j)_{t,i})}
      + Σ_{k=1}^{K-1} e^{g_ν(x^(j)_{t,i-1}, a^(j)_{t+i-1}, z^(k)_{t,i})})] averaged over i, j, t
  update ω_target using exponential moving averaging, ω_target ← αω_target + (1 − α)ω
  update π using R_intrinsic(o_s, a_s, o_{s+1}) := Σ_{t+i=s+1} [(1/λ) R^rec._{θ,η}(x_{t,i-1}, a_{t+i-1}) + R^{K,con.}_{θ,ν}(x_{t,i-1}, a_{t+i-1})]
end repeat
```

time dimension is not used as negatives. For optimization, the Adam optimizer is used with learning rate 10^{-4} and $b_1 = 0.9$ for both the reconstruction loss and contrastive loss. Alternating optimization is used to optimize the critic, with single optimization steps for the critic interleaved with single optimization steps for the rest of the networks. Where necessary to provide multiple input vectors to the reconstructor, generator, and critic networks, inputs are first combined by concatenation before being fed into the first layer of the networks.

Note: Relative to the dimensionality of the "true" source of stochasticity in the world (i.e. within any reparameterized model that accurately captures the world's dynamics), the dimensionality of the generator noise ϵ and hindsight vector Z should not be too small. If it is too small, it may have insufficient capacity to model the stochasticity well, which means there may still be remaining stochastic traps that can negatively affect curiosity-based exploration. On the other hand, making it very large would avoid this problem, but may potentially make learning proceed more slowly. This is because not only does Z have to capture stochasticity, it also has to obey the constraint that it be independent of states and actions, so in very high dimensions this may not be as easy to simultaneously optimize. Of course, the "true" dimensionality of stochasticity is rarely known, so we may make an educated guess when setting sensible values for the noise and hindsight dimensions. In our experiments, we do not tune these dimensions, and simply set it to 256 for all environments and experiments.

D. Discussion and Related Work

D.1. Additional Discussion

It is important to account for stochasticity in reinforcement learning—especially in exploration: Stochasticity may arise naturally due to a variety of factors, such as inherent randomness in the world (e.g. coin flip), or imperfect observations or actions (e.g. faulty sensors or actuators), or un-modeled complexity (e.g. model mismatch or imperfect optimization), or simply due to the existence of other agents (e.g. in a multi-agent game)—all of which would lead to world dynamics that appear stochastic to the agent. In this work, we used a variety of settings to test our hypothesis that hindsight information can mitigate stochastic traps in predictive error-based exploration. Overall, our results verify the fact that learned hindsight representations are able to disentangle the various (unpredictable) stochasticities from the rest of the (predictable) dynamics of the world. Future work may investigate applicability to other scenarios, such as robotics settings, or multi-agent settings.

The invariance objective is reminiscent of that used in counterfactual credit assignment [86], where hindsight information in future-conditional value functions are constrained to not contain information about the agent's actions. A key difference is that in the exploration setting, the invariance constraint primarily serves as a way to ensure that intrinsic rewards do not fall to zero prematurely (i.e. when too much information is leaked about the future), so performance is not so sensitive to violations of the independence constraint—unlike in counterfactual credit assignment, where it takes paramount importance for estimators to be unbiased. Another difference is that hindsight variables are deterministic functions of the future in counterfactual credit assignment, whereas in our case they must be stochastic in order to accommodate the general case of any stochasticity.

Finally, it is worth emphasizing that the world model that is the focus of curiosity-driven exploration as studied in this work (i.e. either the predictive one in `BYOL-Explore`, or the reconstructive one in `BYOL-Hindsight`) is only designed for computing rewards. It need not be related to the underlying RL algorithm, which can be model-free (as is the case for VMPO). While this is a flexibility, it could also be a limitation: Future work may more systematically explore the advantages and disadvantages of sharing learned exploratory world models or representations with the underlying RL algorithm itself. As another remark for clarification, note that "contrastive learning" here refers to a slightly different objective than what is typically referred to in self-supervised representation learning (see e.g. [14]), or in noise contrastive estimation (see e.g. [102]). Indeed, a key innovation in BYOL (and inherited by BYOL-Explore) is that contrastive learning using negative samples are not involved at all. In this work, we start from BYOL-Explore but re-introduce contrastive learning for an orthogonal objective—that is, to learn a hindsight vector for addressing stochastic traps.

Quality of World Model From the perspective of the agent, stochasticity can arise due to a variety of reasons. Inherent randomness in the world is one (e.g. a coin flip), imperfect observations is another (e.g. faulty sensors), and imperfectly executed actions is also another (e.g. faulty actuators). Yet another source of "stochasticity" is that the world model is bad, e.g. due to insufficient capacity to model the real world, or due to imperfect optimization procedure. To the agent, this simply appears that there is remaining "stochasticity" even after long periods of training. Importantly, if there is are particular parts of the world that the model is insufficiently expressive to capture relative to other parts of the world, then they effectively become stochastic traps that the agent may become stuck around. Note that this problem could affect all curiosity-driven exploration methods (as given by Definition 1), and could affect RND, ICM, and BYOL-Explore. Actually, Curiosity in Hindsight should generally potentially *mitigate* this kind of problem, because the hindsight generator will attempt to learn hindsight vectors that capture this otherwise remaining "stochasticity". Now of course, we may then ask the (meta) question: Even with the additional hindsight capacity, what if prediction/reconstruction is still bad, e.g. if the augmented model still has insufficient capacity to model the real world? In general, regardless of whether we are using the augmented model or not, remaining stochasticity is a problem if it is *unevenly* distributed around the world (e.g. high around specific objects, which become traps), but it is not a problem otherwise. So practically, this means the final model needs to be at least "good enough" such that errors are not due to failures to model specific areas of the state space. In our experiments, we empirically observe that reconstruction error becomes close to zero but above zero, but the agents do not get trapped by any remaining unevenly distributed noise. Finally, going forward, it is beneficial for Crafter [103] to be a testbed for curiosity-driven exploration (and curiosity in hindsight) in future updates to this agenda, because it is also a partially observable stochastic environment in which predictive world models may not easily capture all outcomes.

Effects of Two Losses The overall exploration algorithm operates by having the policy parameters maximize the intrinsic rewards, while the model parameters minimize the intrinsic rewards. So, by default the reconstruction loss and contrastive loss affect both the policy and the model. Starting from this base case, we can ask what happens if either loss is omitted, for either policy learning or model learning. *Model Learning*: This is easiest to reason about. On one hand, if the contrastive loss were omitted from model learning, then there is no reason at all for hindsight vectors Z to be independent of X, A, thus Z may very quickly learn to copy all of the information in the outcome itself. This means that reconstruction errors (which yield intrinsic reward for the agent's policy) quickly drops to zero, without the agent having to explore at all. This is disastrous for exploration. On the other hand, if the reconstruction loss were omitted from model learning, then reconstruction errors (which yield intrinsic reward for the agent's policy) will never improve over time no matter how much experience the agent has, which is also pathological for exploration. In sum, to avoid breakdown of exploration, it must be the case that both the reconstruction loss and contrastive loss are applied to model learning. *Policy Learning*: This is also very easy to reason about. We empirically find that in practice, the invariance constraint is always respected very well (see e.g. Figure 15, where the invariance loss remains close to zero throughout training). Therefore the magnitude of the contrastive component of the intrinsic reward always has little to no contribution to the total intrinsic reward. Empirically, indeed exploration behavior is rather unaffected by whether or not the contrastive loss is included as an intrinsic reward for policy learning. On the other hand, suppose we remove the reconstruction component of the intrinsic reward. Then the agent would practically have no incentive to explore, which is disastrous. In sum, for policy learning the scaling between the two intrinsic rewards is not meaningful, as the reconstruction term dominates.

Choice of Baseline Method Briefly, within the curiosity-driven exploration paradigm (Definition 1), there are two primary choices when it comes to representation: How to represent "input states" X_t, and how to represent "target states" X_{t+1}. For the former, we simply make the most general choice of using learned RNN "belief" representations for X_t—that is, rollups of previous actions $\{a_{t'}\}_{t'<t}$ and observation encodings $\{\omega(o_{t'})\}_{t'\leq t}$, where ω is a learned encoding function. This should be uncontroversial, and the vast majority of methods in Table 1 operate this way, since pretty much any non-trivial environment would require the agent to be aware of histories rather than just immediate-state contexts or features. For the latter, we choose BYOL-Explore, which has that "target states" are ℓ_2-normalized encodings of future observations, with the target encoding function ω_{target} being an exponential moving average of ω. This is due to three reasons: First, pixel-based curiosity has been found too perform worse than using learned representations [9, 15]. So, among popular and successful representation methods, we are looking at autoencoded features, random features, inverse dynamics features, and the recently proposed bootstrapped features. Second, moreover, autoencoded features have been found to be unstable in practice [9], and also bootstrapped features have most recently been found to yield the best performance on benchmarks [14]. Third, bootstrapped representations arguably yield the simplest learning algorithm, as the target embedding network is just the exponential moving average of the online embedding network. In sum, this is why we chose BYOL-Explore as the baseline formulation—because it serves a simple and already state-of-the-art baseline on top of which we can demonstrate how Curiosity in Hindsight further improves its performance in stochastic environments.

Generality of Framework The practical framework for Curiosity in Hindsight (as defined by Equations 21–22) can be used to augment any *curiosity-driven* method (as defined by Equations 1–2). As shown in Table 1, this includes a number of recent methods, i.e. as long as they can be expressed in the form of Definition 1. (Moreover, our Response (A.1) above discusses the main reasons why we particularly select BYOL-Explore as our prime example to augment and conduct empirical experiments for). On the other hand, there are other exploration paradigms that *cannot* be expressed in the form of Definition 1 (i.e. not "curiosity-driven" as defined there), and hence cannot be readily augmented with our framework for Curiosity in Hindsight. As discussed in Section 2.2 ("Related Work"), this includes for instance methods based on visitation counts, hashes, and density estimates, as well as methods based on estimated uncertainties about the world, such as taking actions that maximize the estimated information gain, etc. For instance, exploration by the Plan2Explore method operates by first estimating "novelty" through ensemble disagreement in latent predictions made by 1-step transition models, and the agent uses a concurrently trained global recurrent world model to plan to explore on the basis of this novelty measure. So this method operates by directly estimating/maximizing the expected information gain using ensemble disagreement, therefore it does not fall within the curiosity-driven paradigm (i.e. cannot be formulated as an instance of Definition 1), and hence cannot be plugged into Curiosity in Hindsight.

D.2. Additional Related Work

Several related works are concurrent to ours. In the novelty-based family, [104] extends count-based episodic bonuses to continuous state spaces and encourages exploring states that are diverse under a learned embedding, and [105] proposes clustering-based density estimation to model a wide range of timescales. Tackling the problem that stochasticity poses for predictive error-based exploration as we do, [106] is an another extension of `BYOL-Explore` that approaches stochasticity in world dynamics by deliberately and directly leaking a noisy version of future information to the predictor. Theoretically, [107] studies the learning dynamics of self-predictive learning for reinforcement learning, which is related to `BYOL-Explore`. Finally, [108] augments future-conditional supervised learning with the ability to remove uncontrollable information from the future-conditioning variable—which is the “mirror image” of what we seek from hindsight representations in our framework.

References

[1] Nikolaus Kriegeskorte and Pamela K Douglas. Cognitive computational neuroscience. *Nature neuroscience*, 21(9):1148–1160, 2018.

[2] Tianpei Yang, Hongyao Tang, Chenjia Bai, Jinyi Liu, Jianye Hao, Zhaopeng Meng, and Peng Liu. Exploration in deep reinforcement learning: a comprehensive survey. *arXiv preprint arXiv:2109.06668*, 2021.

[3] Jürgen Schmidhuber. A possibility for implementing curiosity and boredom in model-building neural controllers. In *Proc. of the international conference on simulation of adaptive behavior: From animals to animats*, pages 222–227, 1991.

[4] Sebastian Thrun. Exploration in active learning. *Handbook of Brain Science and Neural Networks*, pages 381–384, 1995.

[5] Andrew G Barto, Satinder Singh, Nuttapong Chentanez, et al. Intrinsically motivated learning of hierarchical collections of skills. In *Proceedings of the 3rd International Conference on Development and Learning*, pages 112–19. Piscataway, NJ, 2004.

[6] Junhyuk Oh, Xiaoxiao Guo, Honglak Lee, Richard L Lewis, and Satinder Singh. Action-conditional video prediction using deep networks in atari games. *Advances in neural information processing systems*, 28, 2015.

[7] Chelsea Finn, Ian Goodfellow, and Sergey Levine. Unsupervised learning for physical interaction through video prediction. *Advances in neural information processing systems*, 29, 2016.

[8] Karol Gregor, Danilo Jimenez Rezende, Frederic Besse, Yan Wu, Hamza Merzic, and Aaron van den Oord. Shaping belief states with generative environment models for rl. *Advances in Neural Information Processing Systems*, 32, 2019.

[9] Yuri Burda, Harri Edwards, Deepak Pathak, Amos Storkey, Trevor Darrell, and Alexei A Efros. Large-scale study of curiosity-driven learning. *International Conference on Learning Representations*, 2019.

[10] Bradly C Stadie, Sergey Levine, and Pieter Abbeel. Incentivizing exploration in reinforcement learning with deep predictive models. *International Conference on Learning Representations*, 2016.

[11] Deepak Pathak, Pulkit Agrawal, Alexei A Efros, and Trevor Darrell. Curiosity-driven exploration by self-supervised prediction. In *International conference on machine learning*, pages 2778–2787. PMLR, 2017.

[12] Yuri Burda, Harrison Edwards, Amos Storkey, and Oleg Klimov. Exploration by random network distillation. *International Conference on Learning Representations*, 2019.

[13] Hyoungseok Kim, Jaekyeom Kim, Yeonwoo Jeong, Sergey Levine, and Hyun Oh Song. Emi: Exploration with mutual information. In *International Conference on Machine Learning*, pages 3360–3369. PMLR, 2019.

[14] Zhaohan Daniel Guo, Shantanu Thakoor, Miruna Pîslar, Bernardo Avila Pires, Florent Altché, Corentin Tallec, Alaa Saade, Daniele Calandriello, Jean-Bastien Grill, Yunhao Tang, et al. Byol-explore: Exploration by bootstrapped prediction. *Advances in neural information processing systems*, 35, 2022.

[15] Augustine Mavor-Parker, Kimberly Young, Caswell Barry, and Lewis Griffin. How to stay curious while avoiding noisy tvs using aleatoric uncertainty estimation. In *International Conference on Machine Learning*, pages 15220–15240. PMLR, 2022.

[16] Leshem Choshen, Lior Fox, and Yonatan Loewenstein. Dora the explorer: Directed outreaching reinforcement action-selection. *International Conference on Learning Representations*, 2018.

[17] Laurent Orseau, Tor Lattimore, and Marcus Hutter. Universal knowledge-seeking agents for stochastic environments. In *International conference on algorithmic learning theory*, pages 158–172. Springer, 2013.

[18] Zhang-Wei Hong, Tsu-Jui Fu, Tzu-Yun Shann, and Chun-Yi Lee. Adversarial active exploration for inverse dynamics model learning. In *Conference on Robot Learning*, pages 552–565. PMLR, 2020.

[19] Deepak Pathak, Dhiraj Gandhi, and Abhinav Gupta. Self-supervised exploration via disagreement. In *International conference on machine learning*, pages 5062–5071. PMLR, 2019.

[20] Kuno Kim, Megumi Sano, Julian De Freitas, Nick Haber, and Daniel Yamins. Active world model learning with progress curiosity. In *International conference on machine learning*, pages 5306–5315. PMLR, 2020.

[21] Mikael Henaff. Explicit explore-exploit algorithms in continuous state spaces. *Advances in Neural Information Processing Systems*, 32, 2019.

[22] Pranav Shyam, Wojciech Jaśkowski, and Faustino Gomez. Model-based active exploration. In *International conference on machine learning*, pages 5779–5788. PMLR, 2019.

[23] Ian Osband, Charles Blundell, Alexander Pritzel, and Benjamin Van Roy. Deep exploration via bootstrapped dqn. *Advances in neural information processing systems*, 29, 2016.

[24] Ian Osband, John Aslanides, and Albin Cassirer. Randomized prior functions for deep reinforcement learning. *Advances in Neural Information Processing Systems*, 31, 2018.

[25] Alexander L Strehl and Michael L Littman. An analysis of model-based interval estimation for markov decision processes. *Journal of Computer and System Sciences*, 74, 2008.

[26] Haoran Tang, Rein Houthooft, Davis Foote, Adam Stooke, OpenAI Xi Chen, Yan Duan, John Schulman, Filip DeTurck, and Pieter Abbeel. # exploration: A study of count-based exploration for deep reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

[27] Marc Bellemare, Sriram Srinivasan, Georg Ostrovski, Tom Schaul, David Saxton, and Remi Munos. Unifying count-based exploration and intrinsic motivation. *Advances in neural information processing systems*, 29, 2016.

[28] Georg Ostrovski, Marc G Bellemare, Aäron Oord, and Rémi Munos. Count-based exploration with neural density models. In *International conference on machine learning*, pages 2721–2730. PMLR, 2017.

[29] Rui Zhao and Volker Tresp. Curiosity-driven experience prioritization via density estimation. *Advances in neural information processing systems*, 31, 2018.

[30] Omar Darwiche Domingues, Corentin Tallec, Remi Munos, and Michal Valko. Density-based bonuses on learned representations for reward-free exploration in deep reinforcement learning. In *ICML 2021 Workshop on Unsupervised Reinforcement Learning*, 2021.

[31] Justin Fu, John Co-Reyes, and Sergey Levine. Ex2: Exploration with exemplar models for deep reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

[32] Yannis Flet-Berliac, Johan Ferret, Olivier Pietquin, Philippe Preux, and Matthieu Geist. Adversarially guided actor-critic. *International Conference on Learning Representations*, 2021.

[33] Nikolay Savinov, Anton Raichuk, Raphaël Marinier, Damien Vincent, Marc Pollefeys, Timothy Lillicrap, and Sylvain Gelly. Episodic curiosity through reachability. *International Conference on Learning Representations*, 2019.

[34] Adrià Puigdomènech Badia, Pablo Sprechmann, Alex Vitvitskyi, Daniel Guo, Bilal Piot, Steven Kapturowski, Olivier Tieleman, Martín Arjovsky, Alexander Pritzel, Andew Bolt, et al. Never give up: Learning directed exploration strategies. *International Conference on Learning Representations*, 2020.

[35] Adrià Puigdomènech Badia, Bilal Piot, Steven Kapturowski, Pablo Sprechmann, Alex Vitvitskyi, Zhaohan Daniel Guo, and Charles Blundell. Agent57: Outperforming the atari human benchmark. In *International Conference on Machine Learning*, pages 507–517. PMLR, 2020.

[36] Marlos C Machado, Marc G Bellemare, and Michael Bowling. Count-based exploration with the successor representation. In *ICML Workshop on Exploration in Reinforcement Learning*, 2018.

[37] Min-hwan Oh and Garud Iyengar. Directed exploration in pac model-free reinforcement learning. In *ICML Workshop on Exploration in Reinforcement Learning*, 2018.

[38] Marlos C Machado, Marc G Bellemare, and Michael Bowling. Count-based exploration with the successor representation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2020.

[39] David A Cohn, Zoubin Ghahramani, and Michael I Jordan. Active learning with statistical models. *Journal of artificial intelligence research*, 4:129–145, 1996.

[40] Laurent Itti and Pierre Baldi. Bayesian surprise attracts human attention. *Vision research*, 49(10):1295–1306, 2009.

[41] Mauricio Araya, Olivier Buffet, Vincent Thomas, and Françcois Charpillet. A pomdp extension with belief-dependent rewards. *Advances in neural information processing systems*, 23, 2010.

[42] Yi Sun, Faustino Gomez, and Jürgen Schmidhuber. Planning to be surprised: Optimal bayesian exploration in dynamic environments. In *International conference on artificial general intelligence*, pages 41–51. Springer, 2011.

[43] Susanne Still and Doina Precup. An information-theoretic approach to curiosity-driven reinforcement learning. *Theory in Biosciences*, 131(3):139–148, 2012.

[44] Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. *Advances in neural information processing systems*, 29, 2016.

[45] Ramanan Sekar, Oleh Rybkin, Kostas Daniilidis, Pieter Abbeel, Danijar Hafner, and Deepak Pathak. Planning to explore via self-supervised world models. In *International Conference on Machine Learning*, pages 8583–8592. PMLR, 2020.

[46] Russell Mendonca, Oleh Rybkin, Kostas Daniilidis, Danijar Hafner, and Deepak Pathak. Discovering and achieving goals via world models. *Advances in Neural Information Processing Systems*, 34:24379–24391, 2021.

[47] Jürgen Schmidhuber. Curious model-building control systems. In *Proc. international joint conference on neural networks*, pages 1458–1463, 1991.

[48] Pierre-Yves Oudeyer, Frdric Kaplan, and Verena V Hafner. Intrinsic motivation systems for autonomous mental development. *IEEE transactions on evolutionary computation*, 11(2):265–286, 2007.

[49] Mohammad Gheshlaghi Azar, Bilal Piot, Bernardo Avila Pires, Jean-Bastien Grill, Florent Altché, and Rémi Munos. World discovery models. *arXiv preprint arXiv:1902.07685*, 2019.

[50] Elad Hazan, Sham Kakade, Karan Singh, and Abby Van Soest. Provably efficient maximum entropy exploration. In *International Conference on Machine Learning*, pages 2681–2691. PMLR, 2019.

[51] Hao Liu and Pieter Abbeel. Behavior from the void: Unsupervised active pre-training. *Advances in Neural Information Processing Systems*, 34:18459–18473, 2021.

[52] Zhaohan Daniel Guo, Mohammad Gheshlaghi Azar, Alaa Saade, Shantanu Thakoor, Bilal Piot, Bernardo Avila Pires, Michal Valko, Thomas Mesnard, Tor Lattimore, and Rémi Munos. Geometric entropic exploration. *arXiv preprint arXiv:2101.02055*, 2021.

[53] Denis Yarats, Rob Fergus, Alessandro Lazaric, and Lerrel Pinto. Reinforcement learning with prototypical representations. In *International Conference on Machine Learning*, pages 11920–11931. PMLR, 2021.

[54] Karol Gregor, Danilo Jimenez Rezende, and Daan Wierstra. Variational intrinsic control. *International Conference on Learning Representations*, 2017.

[55] Joshua Achiam, Harrison Edwards, Dario Amodei, and Pieter Abbeel. Variational option discovery algorithms. *arXiv preprint arXiv:1807.10299*, 2018.

[56] Benjamin Eysenbach, Abhishek Gupta, Julian Ibarz, and Sergey Levine. Diversity is all you need: Learning skills without a reward function. *International Conference on Learning Representations*, 2019.

[57] Lisa Lee, Benjamin Eysenbach, Emilio Parisotto, Eric Xing, Sergey Levine, and Ruslan Salakhutdinov. Efficient exploration via state marginal matching. *arXiv preprint arXiv:1906.05274*, 2019.

[58] Víctor Campos, Alexander Trott, Caiming Xiong, Richard Socher, Xavier Giró-i Nieto, and Jordi Torres. Explore, discover and learn: Unsupervised discovery of state-covering skills. In *International Conference on Machine Learning*, pages 1317–1327. PMLR, 2020.

[59] Archit Sharma, Shixiang Gu, Sergey Levine, Vikash Kumar, and Karol Hausman. Dynamics-aware unsupervised discovery of skills. *International Conference on Learning Representations*, 2020.

[60] Kate Baumli, David Warde-Farley, Steven Hansen, and Volodymyr Mnih. Relative variational intrinsic control. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 6732–6740, 2021.

[61] Oliver Groth, Markus Wulfmeier, Giulia Vezzani, Vibhavari Dasagi, Tim Hertweck, Roland Hafner, Nicolas Heess, and Martin Riedmiller. Is curiosity all you need? on the utility of emergent behaviours from curious exploration. *arXiv preprint arXiv:2109.08603*, 2021.

[62] Taehwan Kwon. Variational intrinsic control revisited. *International Conference on Learning Representations*, 2022.

[63] Hao Liu and Pieter Abbeel. Aps: Active pretraining with successor features. In *International Conference on Machine Learning*, pages 6736–6747. PMLR, 2021.

[64] Benjamin Eysenbach, Ruslan Salakhutdinov, and Sergey Levine. The information geometry of unsupervised reinforcement learning. *International Conference on Learning Representations*, 2022.

[65] Michael Laskin, Hao Liu, Xue Bin Peng, Denis Yarats, Aravind Rajeswaran, and Pieter Abbeel. Cic: Contrastive intrinsic control for unsupervised skill discovery. *arXiv preprint arXiv:2202.00161*, 2022.

[66] Marcin Andrychowicz, Filip Wolski, Alex Ray, Jonas Schneider, Rachel Fong, Peter Welinder, Bob McGrew, Josh Tobin, OpenAI Pieter Abbeel, and Wojciech Zaremba. Hindsight experience replay. *Advances in neural information processing systems*, 30, 2017.

[67] Carlos Florensa, David Held, Xinyang Geng, and Pieter Abbeel. Automatic goal generation for reinforcement learning agents. In *International conference on machine learning*, pages 1515–1528. PMLR, 2018.

[68] Ashvin V Nair, Vitchyr Pong, Murtaza Dalal, Shikhar Bahl, Steven Lin, and Sergey Levine. Visual reinforcement learning with imagined goals. *Advances in neural information processing systems*, 31, 2018.

[69] Meng Fang, Tianyi Zhou, Yali Du, Lei Han, and Zhengyou Zhang. Curriculum-guided hindsight experience replay. *Advances in neural information processing systems*, 32, 2019.

[70] Yunzhi Zhang, Pieter Abbeel, and Lerrel Pinto. Automatic curriculum learning through value disagreement. *Advances in Neural Information Processing Systems*, 33:7648–7659, 2020.

[71] Lars Buesing, Theophane Weber, Yori Zwols, Sebastien Racaniere, Arthur Guez, Jean-Baptiste Lespiau, and Nicolas Heess. Woulda, coulda, shoulda: Counterfactually-guided policy search. *arXiv preprint arXiv:1811.06272*, 2018.

[72] Michael Oberst and David Sontag. Counterfactual off-policy evaluation with gumbel-max structural causal models. In *International Conference on Machine Learning*, pages 4881–4890. PMLR, 2019.

[73] Guy Lorberbom, Daniel D Johnson, Chris J Maddison, Daniel Tarlow, and Tamir Hazan. Learning generalized gumbel-max causal mechanisms. *Advances in Neural Information Processing Systems*, 34:26792–26803, 2021.

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

[75] H Francis Song, Abbas Abdolmaleki, Jost Tobias Springenberg, Aidan Clark, Hubert Soyer, Jack W Rae, Seb Noury, Arun Ahuja, Siqi Liu, Dhruva Tirumala, et al. V-mpo: On-policy maximum a posteriori policy optimization for discrete and continuous control. *arXiv preprint arXiv:1909.12238*, 2019.

[76] Thomas Stepleton. The pycolab game engine, 2017. *URL https://github. com/deepmind/pycolab*, 2017.

[77] Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.

[78] Marlos C Machado, Marc G Bellemare, Erik Talvitie, Joel Veness, Matthew Hausknecht, and Michael Bowling. Revisiting the arcade learning environment: Evaluation protocols and open problems for general agents. *Journal of Artificial Intelligence Research*, 61:523–562, 2018.

[79] Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

[80] Michael Tschannen, Josip Djolonga, Paul K Rubenstein, Sylvain Gelly, and Mario Lucic. On mutual information maximization for representation learning. *arXiv preprint arXiv:1907.13625*, 2019.

[81] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *IEEE/CVF conference on computer vision and pattern recognition*, 2020.

[82] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[83] Feng Wang and Huaping Liu. Understanding the behaviour of contrastive loss. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 2495–2504, 2021.

[84] Anna Harutyunyan, Will Dabney, Thomas Mesnard, Mohammad Gheshlaghi Azar, Bilal Piot, Nicolas Heess, Hado P van Hasselt, Gregory Wayne, Satinder Singh, Doina Precup, et al. Hindsight credit assignment. *Advances in neural information processing systems*, 32, 2019.

[85] Chris Nota, Philip Thomas, and Bruno C Da Silva. Posterior value functions: Hindsight baselines for policy gradient methods. In *International Conference on Machine Learning*, pages 8238–8247. PMLR, 2021.

[86] Thomas Mesnard, Théophane Weber, Fabio Viola, Shantanu Thakoor, Alaa Saade, Anna Harutyunyan, Will Dabney, Tom Stepleton, Nicolas Heess, Arthur Guez, et al. Counterfactual credit assignment in model-free reinforcement learning. In *International Conference on Machine Learning*, 2021.

[87] Amy Zhang, Rowan McAllister, Roberto Calandra, Yarin Gal, and Sergey Levine. Learning invariant representations for reinforcement learning without reconstruction. *arXiv preprint arXiv:2006.10742*, 2020.

[88] Ioana Bica, Daniel Jarrett, and Mihaela van der Schaar. Invariant causal imitation learning for generalizable policies. *Advances in Neural Information Processing Systems*, 34:3952–3964, 2021.

[89] Chaochao Lu, José Miguel Hernández-Lobato, and Bernhard Schölkopf. Invariant causal representation learning for generalization in imitation and reinforcement learning. In *ICLR 2022*, 2022.

[90] Christos Louizos, Kevin Swersky, Yujia Li, Max Welling, and Richard Zemel. The variational fair autoencoder. *International Conference on Learning Representations*, 2016.

[91] Daniel Moyer, Shuyang Gao, Rob Brekelmans, Aram Galstyan, and Greg Ver Steeg. Invariant representations without adversarial training. *Advances in Neural Information Processing Systems*, 31, 2018.

[92] Mohammad Lotfollahi, Mohsen Naghipourfar, Fabian J Theis, and F Alexander Wolf. Conditional out-of-sample generation for unpaired data using trvae. *arXiv preprint arXiv:1910.01791*, 2019.

[93] Adam Foster, Árpi Vezér, Craig A Glastonbury, Páidí Creed, Samer Abujudeh, and Aaron Sim. Contrastive mixture of posteriors. In *International Conference on Machine Learning*, pages 6578–6621. PMLR, 2022.

[94] David Barber Felix Agakov. The im algorithm: a variational approach to information maximization. *Advances in neural information processing systems*, 16(320):201, 2004.

[95] Ben Poole, Sherjil Ozair, Aaron Van Den Oord, Alex Alemi, and George Tucker. On variational bounds of mutual information. In *International Conference on Machine Learning*, pages 5171–5180. PMLR, 2019.

[96] Matteo Hessel, Hubert Soyer, Lasse Espeholt, Wojciech Czarnecki, Simon Schmitt, and Hado van Hasselt. Multi-task deep reinforcement learning with popart. In *Proceedings of the AAAI Conference on Artificial Intelligence*, 2019.

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

[98] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European conference on computer vision (ECCV)*, pages 3–19, 2018.

[99] Kyunghyun Cho, Bart Van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. On the properties of neural machine translation: Encoder-decoder approaches. *arXiv preprint arXiv:1409.1259*, 2014.

[100] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *nature*, 518(7540):529–533, 2015.

[101] Hado Van Hasselt, Arthur Guez, and David Silver. Deep reinforcement learning with double q-learning. In *AAAI conference on artificial intelligence*, 2016.

[102] Daniel Jarrett, Ioana Bica, and Mihaela van der Schaar. Time-series generation by contrastive imitation. *Advances in Neural Information Processing Systems*, 34:28968–28982, 2021.

[103] Danijar Hafner. Benchmarking the spectrum of agent capabilities. *arXiv preprint arXiv:2109.06780*, 2021.

[104] Mikael Henaff, Roberta Raileanu, Minqi Jiang, and Tim Rocktäschel. Exploration via elliptical episodic bonuses. *arXiv preprint arXiv:2210.05805*, 2022.

[105] Alaa Saade, Steven Kapturowski, Daniele Calandriello, Charles Blundell, Michal Valko, Pablo Sprechmann, and Bilal Piot. Robust exploration via clustering-based online density estimation. 2023.

[106] Bilal Piot, Zhaohan Daniel Guo, Shantanu Thakoor, and Mohammad Gheshlaghi Azar. Blade: Robust exploration via diffusion models. In *Deep Reinforcement Learning Workshop NeurIPS 2022*, 2022.

[107] Yunhao Tang, Zhaohan Daniel Guo, Pierre Harvey Richemond, Bernardo Ávila Pires, Yash Chandak, Rémi Munos, Mark Rowland, Mohammad Gheshlaghi Azar, Charline Le Lan, Clare Lyle, et al. Understanding self-predictive learning for reinforcement learning. *arXiv preprint arXiv:2212.03319*, 2022.

[108] Mengjiao Yang, Dale Schuurmans, Pieter Abbeel, and Ofir Nachum. Dichotomy of control. *arXiv preprint arXiv:2210.13435*, 2022.