LEDREAMER: BRIDGING THE UNDERSTANDING OF REPRESENTATION COLLAPSE BETWEEN JEPA AND SAMPLE-EFFICIENT WORLD MODELS

Anonymous authors
Paper under double-blind review

ABSTRACT

Joint Embedding Predictive Architectures (JEPAs) efficiently learn representations without pixel-level reconstruction, with recent methods seeking to replace stabilization heuristics such as stop-gradient with the minimization of well-defined objectives. Although the Sketched Isotropic Gaussian Regularizer (SIGReg) prevents representation collapse in continuous JEPAs, its direct application does not account for the factorized structure of discrete categorical latents used by sample-efficient architectures such as DreamerV3. In these latents, shifts applied to the logits of each categorical distribution do not change the resulting probabilities, but can still affect the regularization objective. We introduce LeDreamer, a decoder-free transformer-based world model that addresses this mismatch by applying SIGReg per distribution to pre-sampling categorical logits. We prove that applying SIGReg to the flattened representation, formed by concatenating the logits of all categorical distributions, admits zero-information solutions whose regularization cost decreases as $\mathcal{O}(1/K)$ with the number of categorical distributions K, whereas applying SIGReg independently to each distribution assigns every such solution a positive cost independent of K. Empirically, per-distribution SIGReg prevents marginal representation collapse where flattened regularization fails. We further find that preventing marginal collapse is insufficient for control: in this setting, retaining stop-gradient remains necessary to learn sufficiently action-sensitive dynamics and achieve competitive agent returns. To systematically evaluate these effects, we introduce Atari-5-100k, a statistically selected five-game ablation suite derived from results for 32 recent sample-efficient world models. Across the complete 26-game Atari 100k benchmark, LeDreamer recovers 79.7% of the interquartile mean score of DreamerV3 while training 2.12 times faster, performing on par with recent decoder-free alternatives while training between 2.0 and 3.8 times faster than evaluated decoder-enabled architectures.

1 INTRODUCTION

World models compress an agent's experience into latent representations whose dynamics can be predicted, and on which behavior can be learned (Ha & Schmidhuber, 2018). Sample-Efficient World Models (SEWMs) take this paradigm to its limit: policies are trained entirely inside imagined rollouts, so that capable agents emerge from experience budgets of roughly two hours of gameplay (Kaiser et al., 2024; Micheli et al., 2023; Hafner et al., 2024). To keep their latent representations from collapsing, most SEWMs anchor them with pixel reconstruction (Hafner et al., 2024; Zhang et al., 2023; Micheli et al., 2023; Robine et al., 2023)—a safeguard that also spends capacity on task-irrelevant visual detail and dominates training cost (Robine et al., 2025).

Joint Embedding Predictive Architectures (JEPAs) (Assran et al., 2023) show that reconstruction is not necessary for representation learning, and that line of work has recently gone further: because stabilization heuristics such as the stop-gradient operator do not correspond to the minimization of a well-defined objective (Ponce et al., 2026), LeJEPA replaces them with the Sketched Isotropic Gaussian Regularizer (SIGReg), which provably prevents collapse by enforcing an isotropic Gaussian embedding law (Balestriero & LeCun, 2025), and LeWM carries this recipe to latent world models, albeit offline and reward-free (Maes et al., 2026). This paper asks whether the principled, decoder-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

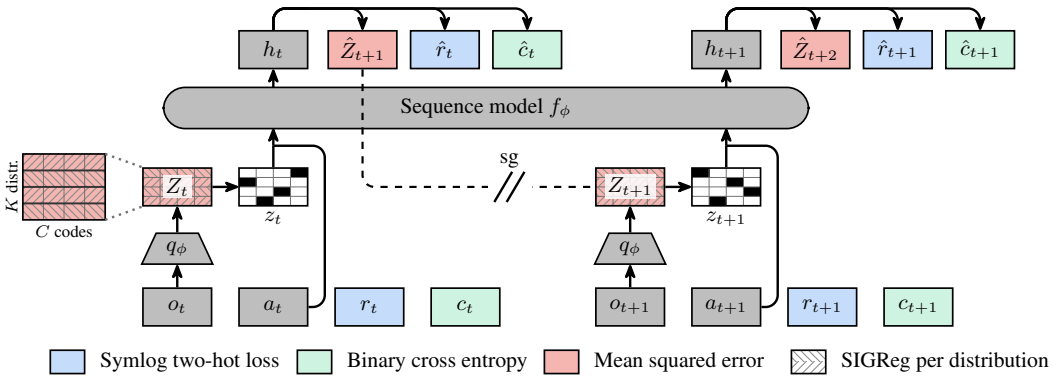

Figure 1: World model self-supervised training in LeDreamer. At each step, the encoder q_ϕ maps the observation o_t to K categorical distributions of C codes each ($\mathcal{Z}_t$), from which a latent z_t is sampled. The action mixer combines z_t with the action a_t, and from the hidden states of the sequence model f_ϕ, prediction heads output the next latent distribution $\hat{\mathcal{Z}}_{t+1}$, the reward $\hat{r}_t$ and the continuation flag $\hat{c}_t$. The target $\mathcal{Z}_{t+1}$ is detached with a stop-gradient (sg). SIGReg is applied independently to the pre-softmax logits of each categorical distribution, indicated by the alternating hatch direction.

free recipe survives the transfer to SEWMs, where representations must be learned online, on non-stationary replayed data, and support policies trained purely in imagination.

We find that the transfer is not a matter of reshaping tensors: the direct transplant fails, and for a structural reason. SEWMs owe part of their sample efficiency to a discrete bottleneck—latents composed of K categorical distributions of C codes each (Hafner et al., 2022; 2024)—a design whose importance our ablations confirm. Categorical probabilities live on the simplex and can never match an isotropic Gaussian target, so SIGReg must act on the pre-softmax logits. This placement opens a gap between what the objectives see: every task loss reads the representation only through the softmax, which is invariant to adding an observation-dependent constant to the logits of each distribution, whereas the regularizer reads the logits directly. The softmax thereby grants the encoder a K-dimensional *gauge* subspace: free channels, invisible to every task loss, into which input-dependent variance can be routed at zero task cost. To study collapse in this setting, we introduce LeDreamer (Figure 1), a decoder-free transformer-based SEWM that applies SIGReg to the categorical logits, and compare the two natural placements of the test: a *flattened* formulation, which concatenates the K blocks into a single vector as in prior JEPA practice, and a *per-distribution* formulation, which tests each categorical distribution independently.

The choice between the two, we prove, is decisive. Through the gauge subspace, an encoder can launder real input variance into directions that no task loss prices, reaching configurations that carry zero information about the observation—and attain exactly zero dynamics loss—yet look near-Gaussian to the flattened test: their regularization cost vanishes as $O(1/K)$, a *fee* that shrinks as the latent gains distributions. The per-distribution test instead *walls* these configurations off, charging every zero-information law a constant penalty $c_0 > 0$ that is independent of K and of anything the encoder controls. Rotated counterexamples show that this power owes nothing to slicing per se: the same degeneracy, rotated out of alignment with the blocks, leaves the per-block test exactly as blind as the flattened one. What matters is *alignment*: the softmax pins its invariance to one known direction per block, and the per-distribution test probes precisely there. The resulting design principle extends beyond our architecture: sketched-isotropy regularization should be applied at the level at which the architecture defines its invariances.

Our experiments confirm this fee-versus-wall picture. Without a stop-gradient, the flattened variant collapses with the exact predicted signature—mutual information at zero, code perplexity pinned at its maximum, and a near-minimal regularizer value that certifies nothing—whereas per-distribution SIGReg preserves informative latents throughout training. Preventing collapse, however, is not the whole story. The regularizer constrains only the marginal law of the latents, while control also depends on their conditional, action-dependent structure. There, unlike recent JEPAs, SEWMs still require the stop-gradient operator used throughout prior work (Zhang et al., 2023; 2026; Robine et al.,

2023; 2025; Burchi & Timofte, 2024; 2025; Hafner et al., 2022; 2024; Micheli et al., 2023; Wang et al., 2025): our action-sensitivity analysis shows that removing it yields dynamics that barely react to actions and agents that fail to reach competitive returns. The bridge between the two families thus carries lessons in both directions: JEPA's principled anti-collapse machinery does transfer to SEWMs once aligned with the architecture's invariances, while the JEPA practice of discarding stop-gradients does not—and our analysis makes precise why, and where the boundary lies.

Finally, we evaluate LeDreamer against the state of the art on the 26-game Atari 100k benchmark. On efficiency the case is unambiguous: LeDreamer trains between 2.0 and 3.8 times faster than every evaluated decoder-enabled world model, and is competitive within the decoder-free family at a fraction of the cost, reaching a median comparable to MuDreamer's (Burchi & Timofte, 2024) in less than half its training time. On performance, the trade is explicit: LeDreamer recovers 79.7% of the interquartile mean of DreamerV3 (Hafner et al., 2024) at less than half its training cost, while the strongest decoder-enabled methods remain ahead under the shared 100k-step budget. The claim of this paper is accordingly not a new state of the art but a favorable operating point and the theory that makes it reachable: collapse prevented by construction, by a provable objective in place of a heuristic.

2 A Decoder-Free World Model with Per-Distribution SIGReg

LeDreamer is a transformer-based sample-efficient world model in the lineage of DreamerV3 (Hafner et al., 2024) and STORM (Zhang et al., 2023), with one deliberate omission and one deliberate addition. The omission is the observation decoder: no reconstruction objective shapes the representation. The addition fills the anti-collapse role that reconstruction implicitly played: an explicit, well-defined regularizer acting on the categorical logits of the latent state. This section presents the components of this design, summarized in Figure 1.

Problem setting. We model the environment as a partially observable Markov decision process (POMDP). At each timestep t, the agent receives an observation $o_t \in \mathbb{R}^{3\times H\times W}$, selects an action a_t from a finite action set $\mathcal{A}$ according to a policy π, and receives a reward $r_t \in \mathbb{R}$ and a continuation flag $c_t \in \{0, 1\}$. Following Ha & Schmidhuber (2018), LeDreamer trains the agent entirely within imagined world model rollouts. Online learning cycles through three steps until the interaction budget is exhausted: (1) collecting transitions into a first-in-first-out (FIFO) replay buffer using the current policy, (2) updating the world model on sampled trajectory segments $(o, a, r, c)_{1:T}$, and (3) improving the policy on rollouts imagined by the model (Hafner et al., 2020). Because the collecting policy π changes throughout training, the replay buffer forms a non-stationary data distribution.

Architecture. LeDreamer models the environment with two main neural networks: an observation encoder and a sequence model. Following prior work (Hafner et al., 2022; Zhang et al., 2023), the convolutional encoder q_ϕ maps each observation o_t to K categorical distributions of C codes each, collectively denoted $\mathcal{Z}_t$. No decoder is attached to this bottleneck: the representation is shaped only by the prediction losses and the regularizer introduced below. A latent sample $z_t \sim \mathcal{Z}_t$, drawn with straight-through gradients (Bengio et al., 2013), is fused with the action a_t by an action mixer m_ϕ:

$$z_t \sim q_\phi(z_t \mid o_t) = \mathcal{Z}_t, \qquad e_t = m_\phi(z_t, a_t). \tag{1}$$

The tokens $e_{1:T}$ are processed by the sequence model f_ϕ, a decoder-only Transformer (Vaswani et al., 2023; Radford & Narasimhan, 2018), and three MLP heads map its hidden states h_t to the next-latent distribution, the reward, and the continuation flag:

$$h_{1:T} = f_\phi(e_{1:T}), \quad \hat{\mathcal{Z}}_{t+1} = g^D_\phi(h_t), \quad \hat{r}_t = g^R_\phi(h_t), \quad \hat{c}_t = g^C_\phi(h_t). \tag{2}$$

Imagination-based agent. For step (3) of the online loop, our agent follows the actor–critic formulation of DreamerV3, in the form used by STORM, which predates the addition of a critic loss on replayed trajectories. The actor maximizes the discounted return of imagined trajectories, while the critic estimates the distribution of those returns with a symlog two-hot loss. Following STORM's official codebase, the critic is additionally regressed toward λ-returns (Sutton & Barto, 2018) recomputed with an exponential moving average of its own parameters, replacing the value regularizer of DreamerV3. Appendix C specifies the imagination procedure and the actor and critic objectives in full; Appendices H and I cover architectures and hyperparameters for all components.

Training objective. The world model is trained end to end in a self-supervised manner on batches of B_1 trajectory segments of length T. The total objective combines a reward loss, a continuation loss, a dynamics loss, and an anti-collapse regularizer applied to the encoder logits:

$$\mathcal{L}(\phi) = \frac{1}{B_1 T} \sum_{n=1}^{B_1} \sum_{t=1}^{T} \Big[\mathcal{L}^{\text{sym}}(\hat{r}_t, r_t) + \mathcal{L}^{\text{bce}}(\hat{c}_t, c_t) + \mathcal{L}^{\text{dyn}}(\hat{\mathcal{Z}}_{t+1}, \mathcal{Z}_{t+1}) \Big] + \lambda_{\text{SIGReg}} \mathcal{R}\big(\ell_{1:T}\big), \quad (3)$$

where $\ell_t^{(k)} \in \mathbb{R}^{B_1 \times C}$ collects the pre-softmax logits of the k-th categorical distribution across the batch, $\ell_t \in \mathbb{R}^{B_1 \times d}$ concatenates the K blocks ($d := KC$), and λ_{SIGReg} weights the anti-collapse regularizer $\mathcal{R}$, whose two instantiations are defined in equation 6. Rewards are supervised with the symlog two-hot loss $\mathcal{L}^{\text{sym}}$ of Hafner et al. (2024), and continuation flags with binary cross entropy $\mathcal{L}^{\text{bce}}$. The dynamics loss $\mathcal{L}^{\text{dyn}}$ matches the predicted prior $\hat{\mathcal{Z}}_{t+1}$ to the encoder's posterior $\mathcal{Z}_{t+1}$, with $\log$ applied elementwise to the categorical probability vectors:

$$\mathcal{L}^{\text{dyn}}(\hat{\mathcal{Z}}_{t+1}, \mathcal{Z}_{t+1}) = \big\| \operatorname{sg}\big(\log \mathcal{Z}_{t+1}\big) - \log \hat{\mathcal{Z}}_{t+1} \big\|_2^2. \quad (4)$$

We adopt this MSE loss on log probabilities in place of standard KL balancing (Hafner et al., 2022) since, in a decoder-free setting, this pulls encoder representations toward sequence model predictions and destabilizes latent features. In our objective, the stop-gradient $\operatorname{sg}(\cdot)$ prevents the encoder from adapting its output to make the prediction target easier to match under non-stationary online data, a strategy also used by SGF (Robine et al., 2025).

Anti-collapse regularization. Without a decoder to ground the representations, nothing in equation 3 prevents the encoder from minimizing the dynamics loss trivially, for instance by mapping every observation to the same latent distribution. We therefore adopt the Sketched Isotropic Gaussian Regularizer (SIGReg) of Balestriero & LeCun (2025). Its univariate building block is an Epps–Pulley statistic that compares a projected law against a *fixed* standard normal target: for a real random variable X with characteristic function φ_X,

$$T_\rho(X) \;=\; \int_{\mathbb{R}} \big|\varphi_X(t) - e^{-t^2/2}\big|^2 \, \mathrm{d}\rho(t), \quad (5)$$

where ρ is a finite nonnegative test weight. Following LeJEPA, no per-projection standardization is applied, and ρ is a 17-knot quadrature approximation of the Gaussian weight (Table 8)—that is, a finite atomic measure; both choices are load-bearing for the analysis (Assumptions 2 and 1). In practice, φ_X is replaced by the empirical characteristic function of the projected batch.

For a batch of B_1 vectors in $\mathbb{R}^m$, the sketched statistic $\mathcal{T}$ projects them onto M random unit directions of $\mathbb{S}^{m-1}$ and averages the empirical T_ρ over those projections. The two instantiations of the regularizer $\mathcal{R}$ in equation 3 apply $\mathcal{T}$ at different levels of the logit tensor:

$$\mathcal{R}^{\text{pd}}\big(\ell_{1:T}\big) \;=\; \frac{1}{TK} \sum_{t=1}^{T} \sum_{k=1}^{K} \mathcal{T}\big(\ell_t^{(k)}\big), \qquad \mathcal{R}^{\text{flat}}\big(\ell_{1:T}\big) \;=\; \frac{1}{T} \sum_{t=1}^{T} \mathcal{T}\big(\ell_t\big). \quad (6)$$

The per-distribution form is our method: where Balestriero & LeCun (2025) compute the statistic over the B embeddings of each of N independent groups, we treat each (timestep, latent) pair as its own group, so a batch contains $T \times K$ distinct categorical distributions, each regularized independently over $\mathbb{S}^{C-1}$ using its own B_1 logit vectors. The *flattened* form—the direct transplant of the one-dimensional JEPA formulation—instead concatenates the K blocks and sketches over $\mathbb{S}^{d-1}$.

For the analysis that follows, we record the population forms of the two variants: writing $\ell \in \mathbb{R}^d$ for the concatenated logits of a single random observation with law P (the time index is dropped) and $\ell^{(k)} \in \mathbb{R}^C$ for its k-th block,

$$S_\rho^{\text{pd}}(P) \;=\; \frac{1}{K} \sum_{k=1}^{K} \mathbb{E}_{v \sim \text{Unif}(\mathbb{S}^{C-1})} T_\rho\big(\langle v, \ell^{(k)} \rangle\big), \qquad S_\rho^{\text{flat}}(P) \;=\; \mathbb{E}_{u \sim \text{Unif}(\mathbb{S}^{d-1})} T_\rho\big(\langle u, \ell \rangle\big). \quad (7)$$

Each variant in equation 6 is the Monte Carlo estimate of its population counterpart in equation 7, with M sampled directions and the empirical characteristic function over B_1 samples.

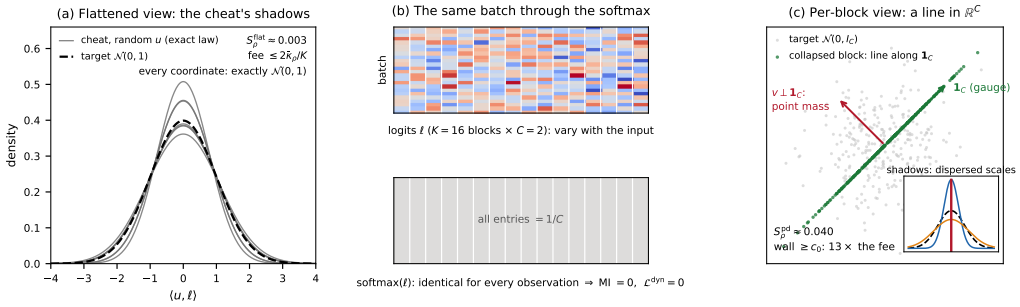

Figure 2: The gauge cheat in miniature (K=16 blocks of C=2 codes). **(a)** Random projections of the flattened logits hug the target $\mathcal{N}(0,1)$: the fee of Theorem 3. **(b)** Every observation yields the same uniform posteriors (Lemma 1). **(c)** Each block is a line along $\mathbf{1}_C$, which no rescaled scalar law can match at every probed scale: the wall of Theorem 3. Closed-form penalties annotated (12.7$\times$ apart).

3 Theoretical Analysis

Preventing collapse in a decoder-free SEWM turns out to be harder than transplanting SIGReg suggests, and the difficulty is created by the discrete bottleneck itself. Categorical latents are central to sample efficiency (Figure 4), but softmax outputs live on the probability simplex and can never match an isotropic Gaussian target, so the regularizer must act on the pre-softmax logits. This opens a structural gap in equation 3: every task loss reads the representation only *through* the softmax, while the regularizer reads the logits directly—and the softmax discards part of what the regularizer sees.

What the softmax discards is exactly a K-dimensional subspace. Since $\mathrm{softmax}(\ell^{(k)} + c\,\mathbf{1}_C) = \mathrm{softmax}(\ell^{(k)})$ (with $\mathbf{1}_C$ the all-ones vector), adding an observation-dependent constant to each block changes no posterior, no sampled code, no dynamics target, and no reward or continuation prediction: every term of equation 3 except the regularizer is invariant (Lemma 1). The K per-block means are therefore free channels in which the encoder can place arbitrary *input-dependent* variance at exactly zero task cost. The same lemma identifies the collapse set exactly: the representation carries no information precisely when the centered logits are constant across observations; there every posterior is uniform, the dynamics target is the same constant vector for every observation and action, and $\mathcal{L}^{\text{dyn}}$ attains exactly zero. Figure 2 shows a toy example of this behavior. Total collapse is thus a zero-cost optimum of the dominant representation-shaping term—reachable unless the regularizer prices it.

The gauge channels hand the encoder a cheat that the flattened test structurally underprices. Set the centered logits to zero and let the K block means carry whitened features of the observation, with total variance matched to the target: the resulting logit law is a trace-matched rank-K Gaussian in dimension $d = KC$ whose scalar coordinates are all *exactly* $\mathcal{N}(0,1)$; the only pathology—perfect intra-block correlation—is invisible to coordinatewise marginals, and random one-dimensional projections average it away (a typical direction sees variance $1 \pm \sqrt{2/K}$; Lemma 2). This is precisely how the flattened variant fails in practice (Figure 3): mutual information drops to zero while code perplexity sits at its maximum C—the logits keep varying, the probabilities carry nothing.

Theorem (fee versus wall; informal). *Let $\mathcal{C}_0$ denote the zero-information manifold: logit laws whose centered logits are constant across observations. Let S_ρ^{flat} and S_ρ^{pd} be the population regularizers of equation 7. Then, in population, i.e., in the limit of infinitely many batch elements and projections,*

$$\inf_{P \in \mathcal{C}_0} S_\rho^{\text{flat}}(P) \;\leq\; \frac{2\bar{\kappa}_\rho}{K}, \qquad \text{whereas} \qquad S_\rho^{\text{pd}}(P) \;\geq\; c_0 \;>\; 0 \quad \text{for every } P \in \mathcal{C}_0, \tag{8}$$

where $\bar{\kappa}_\rho = \frac{1}{4}\int t^4\, d\rho$ depends only on the test weight ($\bar{\kappa}_\rho = 3/4$ for the Gaussian weight) and $c_0 = c_0(\rho, C)$ is explicit and independent of K—and of everything the encoder controls: the lower bound holds for arbitrary laws of the gauge coordinates, with no moment assumptions. Formal statement and proof: Theorem 3 in Appendix J.

The two bounds read as a fee and a wall (Corollary 4). Under the flattened objective, total collapse is available at extra cost at most $2\lambda_{\text{SIGReg}}\bar{\kappa}_\rho/K$, so every configuration whose dynamics loss

alone exceeds that threshold is strictly beaten by collapse; at our configuration $(K, C, \lambda_{\mathrm{SIGReg}}) = (16, 96, 0.1)$ the threshold is at most $\lambda_{\mathrm{SIGReg}} \cdot 0.094$, and the penalty the realized cheat actually pays is measured at $\approx \lambda_{\mathrm{SIGReg}} \cdot 0.0055 = 5.5 \times 10^{-4}$—below the empirical null floor of the statistic itself, which is why enlarging the batch or the projection count does not rescue the flattened variant (Remark 8; Figure 14). Under the per-distribution objective, every collapsed law instead costs at least $\lambda_{\mathrm{SIGReg}} c_0$, and the collapsed law that training realizes is measured at $\approx \lambda_{\mathrm{SIGReg}} \cdot 0.063$—an order-$K$ separation of roughly $11\times$ from the fee it pays under flattening—so any configuration whose dynamics error plus regularizer excess falls below that price is strictly preferred to it.

We note that per-distribution regularization is not vacuously safe. Unlike in LeJEPA (Balestriero & LeCun, 2025), where the task-free uninformative configurations are essentially constant embeddings—point masses, rejected outright by the fixed target—a categorical block can collapse while its logits remain non-degenerate: any law supported on the diagonal $\{c\,\mathbf{1}_C : c \in \mathbb{R}\}$ yields identical posteriors for every observation, yet with a Gaussian coefficient every single logit coordinate is exactly $\mathcal{N}(0, 1)$. Blockwise, the collapsed law is a (translated) line in $\mathbb{R}^C$—support dimension one—whose one-dimensional shadows have χ^2_1-dispersed scales, and the encoder is free to choose *any* law for the diagonal coefficient in the hope of fooling the test. Theorem 3(b) closes this loophole uniformly: no such law exists, because the modulus of the characteristic function removes the constant offset and the fixed target, probed at two separated scales, cannot be matched by a single rescaled scalar law. Both ingredients are load-bearing (Assumptions 2 and 1): a per-projection-standardized variant, or a single-scale test weight, admits collapsed laws with vanishing penalty.

The power of the per-distribution test does not come from slicing per se: rotating the very same rank-K degeneracy out of alignment with the blocks leaves every block marginal a trace-matched rank-K Gaussian, and the per-block test becomes exactly as blind as the flattened one (Proposition 5). The power comes from the softmax pinning its invariance to one *known* direction per block, so that the test is sliced exactly where the architecture's blind spot lives. The design principle we take from this analysis: sketched-isotropy regularization must be applied at the level at which the architecture defines its invariances—for categorical latents, one test per distribution. Formal statements, proofs, explicit constants, and numerical verification are collected in Appendix J.

4 Evaluating Representation Collapse

This section empirically tests the predictions of Section 3: under the flattened regularizer the gauge cheat is affordable and LeDreamer should collapse, whereas the per-distribution wall should preserve informative latents—with or without the stop-gradient heuristic. We compare both formulations, tracking collapse metrics and downstream returns, on **Atari-5-100k**. Since ablating all 26 Atari 100k games over multiple seeds is computationally prohibitive, we adapt the Atari-5 distillation of Aitchison et al. (2022) to the 100k regime, using published results from 32 SEWMs to select the five games—*Bank Heist*, *Breakout*, *James Bond*, *Kangaroo*, and *Kung-Fu Master*—that best predict the full-benchmark IQM (LOOCV $R^2 = 0.922$; details in Appendix E).

We train LeDreamer under six configurations—{SIGReg-Off, Flattened, Per-Distribution} $\times$ {with, without stop-gradient (SG)}—plus, for returns, a *1D-latent* variant that replaces the categorical bottleneck with a continuous embedding regularized as in LeWM (Maes et al., 2026). Collapse is tracked by two metrics: (1) *Normalized mutual information* $I(O; Z_k)$ (Kreer, 1957) vanishes exactly on the zero-information manifold $\mathcal{C}_0$ (Lemma 1) and (2) *code perplexity* (Jelinek et al., 1977) identifies the *form* of a collapse: near 1 for concentration on a few codes, at its maximum C for uniform posteriors. Since perplexity alone cannot separate healthy diversity from uniform collapse (Remark 7), the two metrics are read jointly.

We begin with the cleanest setting—no stop-gradient—where the regularizer alone must prevent collapse. Flattened/SG-Off collapses completely (Figure 3): mutual information drops to zero while perplexity climbs to its maximum C—jointly, the exact fingerprint Theorem 3(a) predicts for the gauge cheat (Remark 2). Per-Distribution/SG-Off, in contrast, retains positive mutual information throughout training: the wall of Theorem 3(b) at work. Disabling SIGReg while keeping the stop-gradient yields the opposite failure, perplexity dropping toward 1: neither the task losses nor the stop-gradient alone maintains diversity (cf. Remark 1). The rightmost panel reports the SIGReg statistic itself: every configuration drives it comparably low—including the collapsed flattened runs. This is not an optimization failure but the fee of Theorem 3(a) realized: under flattening, a low

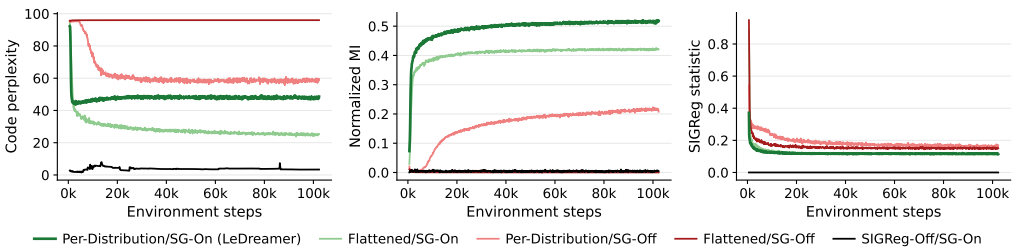

Figure 3: Code perplexity, normalized Mutual Information (MI) between observations and latents, and the SIGReg statistic, averaged over the Atari-5-100k games.

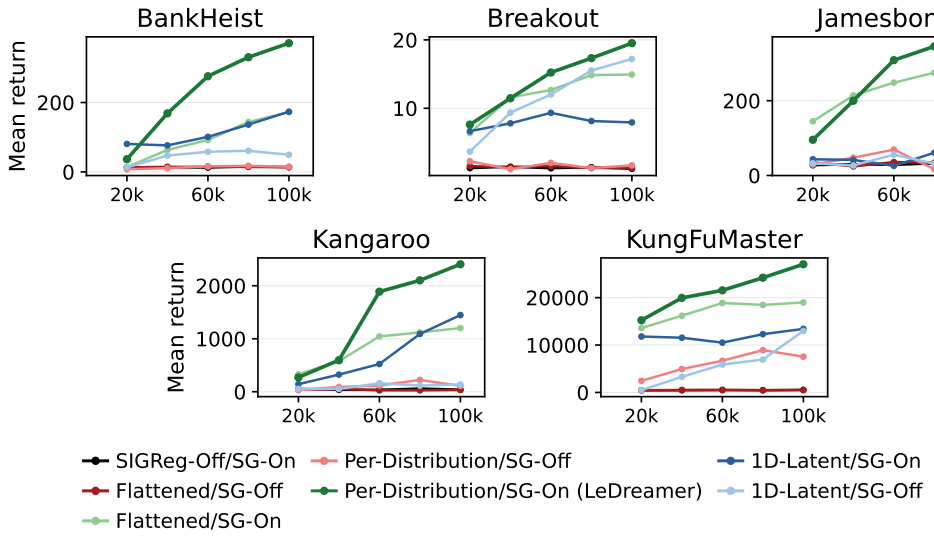

Figure 4: Mean episodic returns on the Atari-5-100k benchmark for each configuration of the representation collapse experiment, averaged over three seeds per configuration (five for LeDreamer).

statistic certifies nothing about diversity, whereas under per-distribution the same low values are reachable only away from the collapse manifold.

Enabling the stop-gradient prevents total collapse under both regularizers—expected, as it is the standard anti-collapse heuristic of prior SEWMs (Hafner et al., 2022; Zhang et al., 2023; Robine et al., 2023; 2025; Burchi & Timofte, 2024). Downstream returns separate the configurations further (Figure 4). First, note that the discrete bottleneck matters: SIGReg on a continuous 1D embedding, as in LeWM, performs far below its categorical counterparts, with or without the stop-gradient. Among the categorical variants, the stop-gradient improves returns under both regularizers, and Per-Distribution with it—LeDreamer—is the strongest by a wide margin. Per-Distribution without it avoids collapse yet still underperforms, a gap traced to reduced action sensitivity (Appendix B): the theory governs only the marginal law of the latents (Remark 9), while the stop-gradient protects the conditional, action-dependent structure that control requires.

Finally, the theory prices the rescue that rescaling $\lambda_{\mathrm{SIGReg}}$ offers: collapse wins whenever $\lambda_{\mathrm{SIGReg}}$ times the regularizer's price of the collapsed law falls below the dynamics savings, so each variant has a critical weight and, assuming comparable dynamics savings, their ratio should be of order K: ≈ 11.5, the ratio between the per-distribution and flattened prices of the collapsed law realized in training (Remarks 8 and 5). Figure 5 tests the prediction on Kung-Fu Master. Both variants exhibit a sharp threshold, below which collapse carries the gauge signature at every sub-critical $\lambda_{\mathrm{SIGReg}}$: ${\lambda_{\mathrm{SIGReg}}}^{*}_{\mathrm{pd}} \in (0.05, 0.08]$ against ${\lambda_{\mathrm{SIGReg}}}^{*}_{\mathrm{flat}} \in (1.0, 1.2]$—a measured ratio in $[12.5, 24]$, of the predicted order. Above their thresholds, both variants plateau near the mutual information of the

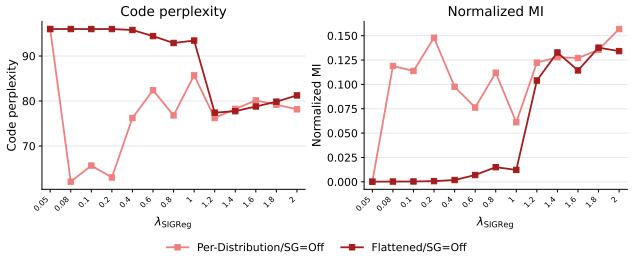

Figure 5: Collapse threshold versus regularization weight on Kung-Fu Master (SG-Off; one seed per point). Below a critical λ_{SIGReg}, each variant collapses with the gauge signature—code perplexity at C (left), zero mutual information (right)—with thresholds an order of magnitude apart.

isotropic target itself ($m^*_{96} \approx 0.106$; Remark 7): in the regularizer-dominated regime, rescaling buys conformity to the target, not more information.

5 Comparison with State-of-the-Art Methods

Sections 3 and 4 established the core claim of this work: per-distribution SIGReg brings principled anti-collapse regularization to SEWMs, where the flattened adaptation of prior JEPA practice provably and empirically fails. This section evaluates LeDreamer against the state of the art, showing that it trains significantly faster than decoder-enabled models and remains competitive with recent decoder-free alternatives, though a performance gap to the strongest decoder-enabled architectures remains, which we quantify below.

Evaluations are conducted on the standard Atari-100k benchmark (Kaiser et al., 2024), which restricts training to 100k environment steps across 26 games. We report human-normalized scores averaged over 5 seeds with 100 evaluation episodes at the final checkpoint. Baselines span three categories: decoder-enabled architectures (DreamerV3 (Hafner et al., 2024), STORM (Zhang et al., 2023), DramaXS (Wang et al., 2025)), decoder-free methods (MuDreamer (Burchi & Timofte, 2024), SGF (Robine et al., 2025)), and EfficientZero-V2 (Wang et al., 2024) as a reference for lookahead search. Figure 6 visualizes IQM confidence intervals alongside median scores relative to training time.

LeDreamer demonstrates high computational efficiency, training between 2.0 and 3.8 times faster than every evaluated decoder-enabled architecture while remaining competitive within the decoder-free category. Specifically, it achieves a higher median than SGF and a median comparable to MuDreamer while training 2.17 times faster. Against DramaXS, a decoder-enabled model, LeDreamer obtains a higher IQM point estimate (0.396 against 0.343) while training 3.8 times faster, although the bootstrap confidence intervals overlap and we therefore regard their performance as competitive.

A performance gap remains relative to the strongest decoder-enabled methods: LeDreamer recovers 79.7% of the IQM of DreamerV3 while training 2.12 times faster, and 62.3% of that of STORM at half the training cost. Under the normalized mean (a metric dominated by environments where agents score far above human level, making it sensitive to outliers), LeDreamer scores 0.768 compared to 0.884 for SGF, making its mean the lowest among all compared methods. Because marginal collapse is prevented by a well-defined objective, closing this remaining gap is no longer a matter of training stability, but of representation richness. A natural next step is to apply similar principled regularization to the conditional, action-dependent structure that still relies on the stop-gradient (Appendix B). For further details on the benchmark protocol, complete per-game scores, and aggregate human-normalized mean, median, and IQM, see Appendix D.

Ablations. We validate our design choices through extensive architectural ablations on the Atari-5-100k benchmark, with comprehensive sweeps and learning curves documented in Appendix F. Across these evaluations, our dynamics objective using MSE on log probabilities achieves higher returns than standard KL balancing. Furthermore, the per-distribution advantage generalizes across alternative regularizers, improving methods like VICReg (Bardes et al., 2022) over flattened variants while outperforming implicit mechanisms like batch normalization. Finally, batch size sweeps indicate that this gain is rooted in the structural constraint of the architecture rather than effective sample size.

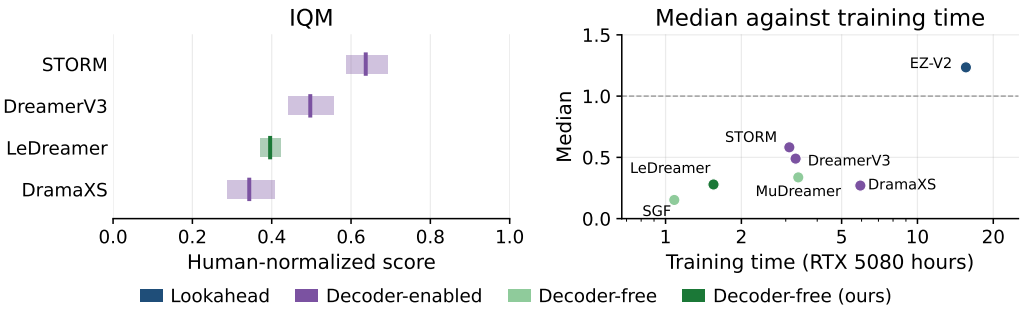

Figure 6: Atari 100k results. **Left:** interquartile mean (IQM) of human-normalized scores with 95% bootstrap confidence intervals, computed over individual runs following Agarwal et al. (2022). SGF, EZ-V2 and MuDreamer are omitted, as the per-seed scores this estimator requires are not publicly available. LeDreamer is competitive with the decoder-enabled methods shown, its interval overlaps that of DramaXS. **Right:** median human-normalized score against training time, measured on a single RTX 5080. LeDreamer attains a median comparable to MuDreamer at less than half the training cost.

6 Related Work

Sample-Efficient World Models. Sample-efficient world models (SEWMs) train policies entirely within imagined rollouts to minimize real environment interactions (Ha & Schmidhuber, 2018). While the Dreamer lineage (Hafner et al., 2020; 2022; 2024) and STORM (Zhang et al., 2023) prevent collapse through pixel reconstruction, recent work pursues decoder-free alternatives. MuDreamer (Burchi & Timofte, 2024) avoids reconstruction via batch normalization and auxiliary prediction tasks, while SGF (Robine et al., 2025) learns deterministic latents with VICReg (Bardes et al., 2022). Concurrently, R2-Dreamer (Morihira et al., 2026) and NE-Dreamer (Bredis et al., 2026) combine Barlow Twins regularization with stop-gradients. However, both target continuous control and update critics on replayed transitions, whereas LeDreamer evaluates discrete control purely within imagined rollouts.

Joint-Embedding Architectures and Regularization. Joint Embedding Predictive Architectures (JEPAs) (Assran et al., 2023) discard pixel reconstruction to predict abstract representations directly, historically relying on heuristics like stop-gradients to prevent collapse. To replace heuristics with theoretical guarantees, LeJEPA (Balestriero & LeCun, 2025) introduced the Sketched Isotropic Gaussian Regularizer (SIGReg). LeWM (Maes et al., 2026) demonstrated that latent dynamics can be trained entirely without stabilization heuristics, though its scope remains restricted to offline, reward-free planning. LeDreamer bridges these paradigms by adapting SIGReg to structured categorical latents in active, online reinforcement learning.

7 Conclusion

We presented LeDreamer, a decoder-free transformer-based sample-efficient world model whose stability rests on a provable objective rather than a heuristic. Collapse in this setting is governed by the gauge invariance of the softmax: flattened SIGReg prices total collapse at a vanishing $O(1/K)$ fee, whereas per-distribution SIGReg charges every zero-information configuration a constant, K-independent penalty—a separation our experiments confirm, including the predicted ratio of critical regularization weights. That guarantee, however, concerns only the marginal law of the latents: unlike in recent JEPAs, the stop-gradient remains necessary to preserve the conditional, action-dependent structure that control requires. On the complete Atari 100k benchmark, LeDreamer recovers 79.7% of DreamerV3's interquartile mean at less than half its training cost—training between 2.0× and 3.8× faster than every evaluated decoder-enabled model—with ablations made affordable by Atari-5-100k, our reproducible five-game suite. The same machinery points forward: regularizing the conditional law directly is the natural route to removing the stop-gradient entirely, and the broader lesson travels—sketched-isotropy regularization belongs at the level at which the architecture defines its invariances.

AI USE STATEMENT

In this work, we used generative AI tools, specifically large language models, throughout the research and writing process, under author direction and with author verification. Among tasks with required disclosure, we used generative AI tools to help develop the theoretical analysis: formulating and refining mathematical claims, providing critical ingredients for proofs, and assisting in the writing and simplification of proofs (including the explicit-constant proof of the lower bound in Theorem 3); to propose and refine hypotheses (for instance, the predicted ratio between the critical regularization weights of the two SIGReg variants); to design and provide feedback on research methodology and experiments (for instance, the selection of sweep values in Figure 5); to implement scripts for experimental and analysis procedures, such as the action-sensitivity analysis; and to interpret experimental results in collaboration with the authors. The identification of the research problem and the design of the proposed method are the authors' own, and all experiments were executed by the authors. Generating synthetic datasets, cleaning and reformatting datasets, supporting qualitative and thematic data analysis, and assisting with translation are not applicable to this work. Among tasks with recommended disclosure, we used generative AI tools for drafting parts of the paper from author-provided outlines, editing text for readability and notational consistency, generating code to produce scientific figures, suggesting experimental parameters, identifying gaps in our ablations, and identifying relevant prior work and literature. We reviewed and verified all AI-assisted work: every mathematical definition, statement, and proof was checked line by line by the authors, with quantitative claims additionally verified numerically against closed forms and Monte Carlo estimates; AI-generated plotting and experimental code was reviewed and tested; and all AI-drafted or AI-edited text was revised by the authors, with unfounded claims identified and removed. We take full responsibility for the final content of this work, including text, claims, and artifacts produced with the aid of generative AI tools.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, we provide the source code for LeDreamer, the scripts used to derive our proposed Atari-5-100k benchmark, and the curated dataset of recent sample-efficient world models used in our benchmark analysis. All released code and datasets are available at `https://anonymous.4open.science/r/DF-TWM-7025/README.md`.

REFERENCES

Pranav Agarwal, Sheldon Andrews, and Samira Ebrahimi Kahou. Learning to play atari in a world of tokens, 2024. URL https://arxiv.org/abs/2406.01361.

Rishabh Agarwal, Max Schwarzer, Pablo Samuel Castro, Aaron Courville, and Marc G. Bellemare. Deep reinforcement learning at the edge of the statistical precipice, 2022. URL https://arxiv.org/abs/2108.13264.

Matthew Aitchison, Penny Sweetser, and Marcus Hutter. Atari-5: Distilling the arcade learning environment down to five games, 2022. URL https://arxiv.org/abs/2210.02019.

Eloi Alonso, Adam Jelley, Vincent Micheli, Anssi Kanervisto, Amos Storkey, Tim Pearce, and François Fleuret. Diffusion for world modeling: Visual details matter in atari, 2024. URL https://arxiv.org/abs/2405.12399.

Mahmoud Assran, Quentin Duval, Ishan Misra, Piotr Bojanowski, Pascal Vincent, Michael Rabbat, Yann LeCun, and Nicolas Ballas. Self-supervised learning from images with a joint-embedding predictive architecture, 2023. URL https://arxiv.org/abs/2301.08243.

Jimmy Lei Ba, Jamie Ryan Kiros, and Geoffrey E. Hinton. Layer normalization, 2016. URL https://arxiv.org/abs/1607.06450.

Randall Balestriero and Yann LeCun. Lejepa: Provable and scalable self-supervised learning without the heuristics, 2025. URL https://arxiv.org/abs/2511.08544.

Adrien Bardes, Jean Ponce, and Yann LeCun. Vicreg: Variance-invariance-covariance regularization for self-supervised learning, 2022. URL https://arxiv.org/abs/2105.04906.

M. G. Bellemare, Y. Naddaf, J. Veness, and M. Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013. ISSN 1076-9757. doi: 10.1613/jair.3912. URL http://dx.doi.org/10.1613/jair.3912.

Yoshua Bengio, Nicholas Léonard, and Aaron Courville. Estimating or propagating gradients through stochastic neurons for conditional computation, 2013. URL https://arxiv.org/abs/1308.3432.

George Bredis, Nikita Balagansky, Daniil Gavrilov, and Ruslan Rakhimov. Next embedding prediction makes world models stronger, 2026. URL https://arxiv.org/abs/2603.02765.

Maxime Burchi and Radu Timofte. Mudreamer: Learning predictive world models without reconstruction, 2024. URL https://arxiv.org/abs/2405.15083.

Maxime Burchi and Radu Timofte. Learning transformer-based world models with contrastive predictive coding, 2025. URL https://arxiv.org/abs/2503.04416.

Maxime Burchi and Radu Timofte. Accurate and efficient world modeling with masked latent transformers, 2026. URL https://arxiv.org/abs/2507.04075.

Lior Cohen, Kaixin Wang, Bingyi Kang, and Shie Mannor. Improving token-based world models with parallel observation prediction. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=Lfp5Dk1xb6.

Lior Cohen, Kaixin Wang, Bingyi Kang, Uri Gadot, and Shie Mannor. Simulus: Combining improvements in sample-efficient world model agents, 2026. URL https://arxiv.org/abs/2502.11537.

Persi Diaconis and David Freedman. Asymptotics of graphical projection pursuit. *The annals of statistics*, pp. 793–815, 1984.

Stefan Elfwing, Eiji Uchibe, and Kenji Doya. Sigmoid-weighted linear units for neural network function approximation in reinforcement learning, 2017. URL https://arxiv.org/abs/1702.03118.

David Ha and Jürgen Schmidhuber. World models. 2018. doi: 10.5281/ZENODO.1207631. URL https://zenodo.org/record/1207631.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination, 2020. URL https://arxiv.org/abs/1912.01603.

Danijar Hafner, Timothy Lillicrap, Mohammad Norouzi, and Jimmy Ba. Mastering atari with discrete world models, 2022. URL https://arxiv.org/abs/2010.02193.

Danijar Hafner, Jurgis Pasukonis, Jimmy Ba, and Timothy Lillicrap. Mastering diverse domains through world models, 2024. URL https://arxiv.org/abs/2301.04104.

F. Jelinek, R. L. Mercer, L. R. Bahl, and J. K. Baker. Perplexity—a measure of the difficulty of speech recognition tasks. *The Journal of the Acoustical Society of America*, 62(S1):S63–S63, December 1977. ISSN 1520-8524. doi: 10.1121/1.2016299. URL http://dx.doi.org/10.1121/1.2016299.

Lukasz Kaiser, Mohammad Babaeizadeh, Piotr Milos, Blazej Osinski, Roy H Campbell, Konrad Czechowski, Dumitru Erhan, Chelsea Finn, Piotr Kozakowski, Sergey Levine, Afroz Mohiuddin, Ryan Sepassi, George Tucker, and Henryk Michalewski. Model-based reinforcement learning for atari, 2024. URL https://arxiv.org/abs/1903.00374.

Youngin Kim, Ray Sun, Inho Kim, Bumsoo Park, and Hyun Oh Song. Identifiable token correspondence for world models, 2026. URL https://arxiv.org/abs/2605.16457.

J. Kreer. A question of terminology. *IRE Transactions on Information Theory*, 3(3):208–208, 1957. doi: 10.1109/TIT.1957.1057418.

Jia-Hua Lee, Bor-Jiun Lin, Wei-Fang Sun, and Chun-Yi Lee. Edeline: Enhancing memory in diffusion-based world models via linear-time sequence modeling, 2025. URL https://arxiv.org/abs/2502.00466.

Tianle Li, Peixi Peng, Qingyu Liu, and Luntong Li. PO-dreamer: Memory guided world models for partially observable reinforcement learning, 2026. URL https://openreview.net/forum?id=QklhZ70C49.

Jing Yu Lim, Rushi Shah, Zarif Ikram, Samson Yu, Haozhe Ma, Tze-Yun Leong, and Dianbo Liu. Jedi: Joint embedding diffusion world model for online model-based reinforcement learning, 2026. URL https://arxiv.org/abs/2605.13013.

Haoyu Ma, Jialong Wu, Ningya Feng, Chenjun Xiao, Dong Li, Jianye Hao, Jianmin Wang, and Mingsheng Long. Harmonydream: Task harmonization inside world models, 2024. URL https://arxiv.org/abs/2310.00344.

Lucas Maes, Quentin Le Lidec, Damien Scieur, Yann LeCun, and Randall Balestriero. Leworldmodel: Stable end-to-end joint-embedding predictive architecture from pixels, 2026. URL https://arxiv.org/abs/2603.19312.

Paul Mattes, Rainer Schlosser, and Ralf Herbrich. Hieros: Hierarchical imagination on structured state space sequence world models, 2024. URL https://arxiv.org/abs/2310.05167.

Cristian Meo, Mircea Lica, Zarif Ikram, Akihiro Nakano, Vedant Shah, Aniket Rajiv Didolkar, Dianbo Liu, Anirudh Goyal, and Justin Dauwels. Masked generative priors improve world models sequence modelling capabilities, 2025. URL https://arxiv.org/abs/2410.07836.

Vincent Micheli, Eloi Alonso, and François Fleuret. Transformers are sample-efficient world models, 2023. URL https://arxiv.org/abs/2209.00588.

Vincent Micheli, Eloi Alonso, and François Fleuret. Efficient world models with context-aware tokenization, 2024. URL https://arxiv.org/abs/2406.19320.

Naoki Morihira, Amal Nahar, Kartik Bharadwaj, Yasuhiro Kato, Akinobu Hayashi, and Tatsuya Harada. R2-dreamer: Redundancy-reduced world models without decoders or augmentation, 2026. URL https://arxiv.org/abs/2603.18202.

Zhao-Han Peng, Shaohui Li, Zhi Li, Shulan Ruan, Yu Liu, and You He. From observations to events: Event-aware world model for reinforcement learning, 2026. URL https://arxiv.org/abs/2601.19336.

Jean Ponce, Basile Terver, Martial Hebert, and Michael Arbel. Dual perspectives on non-contrastive self-supervised learning. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=f5MC1G6XhB.

Alec Radford and Karthik Narasimhan. Improving language understanding by generative pre-training. 2018. URL https://api.semanticscholar.org/CorpusID:49313245.

Jan Robine, Marc Höftmann, Tobias Uelwer, and Stefan Harmeling. Transformer-based world models are happy with 100k interactions, 2023. URL https://arxiv.org/abs/2303.07109.

Jan Robine, Marc Höftmann, and Stefan Harmeling. Simple, good, fast: Self-supervised world models free of baggage, 2025. URL https://arxiv.org/abs/2506.02612.

Richard S. Sutton and Andrew G. Barto. *Reinforcement Learning: An Introduction*. The MIT Press, second edition, 2018. URL http://incompleteideas.net/book/the-book-2nd.html.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need, 2023. URL https://arxiv.org/abs/1706.03762.

Shengjie Wang, Shaohuai Liu, Weirui Ye, Jiacheng You, and Yang Gao. Efficientzero v2: Mastering discrete and continuous control with limited data, 2024. URL https://arxiv.org/abs/2403.00564.

Wenlong Wang, Ivana Dusparic, Yucheng Shi, Ke Zhang, and Vinny Cahill. Drama: Mamba-enabled model-based reinforcement learning is sample and parameter efficient, 2025. URL https://arxiv.org/abs/2410.08893.

Ronald J. Williams. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Mach. Learn.*, 8(3–4):229–256, May 1992. ISSN 0885-6125. doi: 10.1007/BF00992696. URL https://doi.org/10.1007/BF00992696.

Yuxin Wu and Kaiming He. Group normalization, 2018. URL https://arxiv.org/abs/1803.08494.

Boxuan Zhang, Runqing Wang, Wei Xiao, Weipu Zhang, Jian Sun, Gao Huang, Jie Chen, and Gang Wang. Dymodreamer: World modeling with dynamic modulation, 2025. URL https://arxiv.org/abs/2509.24804.

Weipu Zhang, Gang Wang, Jian Sun, Yetian Yuan, and Gao Huang. Storm: Efficient stochastic transformer based world models for reinforcement learning, 2023. URL https://arxiv.org/abs/2310.09615.

Weipu Zhang, Adam Jelley, Trevor McInroe, Amos Storkey, and Gang Wang. Object-centric world models from few-shot annotations for sample-efficient reinforcement learning, 2026. URL https://arxiv.org/abs/2501.16443.

A Representation collapse metrics

Mutual information. Mutual information measures the amount of information that two random variables share, and is zero exactly when they are independent. We take the two variables to be the observation O and the code Z_k sampled from the k-th categorical latent. The code is a codification of the observation, so the two should share information by construction. If they share none, the code carries nothing about the frame it was computed from, the encoder has stopped responding to its input, and the representation has collapsed. Measuring $I(O; Z_k)$ therefore tests directly for the failure mode we are concerned with.

Computation. A batch contains $N = B \cdot T$ observations, and for each one the encoder gives the probabilities of latent k directly, $p_n^{(k)} = \mathrm{softmax}\left(\ell_n^{(k)}\right) \in [0,1]^C$. We compute the mutual information as

$$I\big(O; Z_k\big) \;=\; \underbrace{H\left(\frac{1}{N}\sum_{n=1}^{N} p_n^{(k)}\right)}_{H(Z_k)} \;-\; \underbrace{\frac{1}{N}\sum_{n=1}^{N} H\big(p_n^{(k)}\big)}_{H(Z_k|O)}. \tag{9}$$

The first term is the entropy of the average of the N probability vectors, and the second is the average of their entropies: both use the same probabilities and differ only in whether the average over observations is taken before or after the entropy. We divide by $\log C$ to obtain a value in $[0,1]$ and average over the K distributions.

Code perplexity. Mutual information detects collapse but does not distinguish between its forms, since it falls to zero whenever the code stops depending on the observation, regardless of how. We therefore report code perplexity alongside it, which measures how many codes the latent uses across the dataset. Two failure modes are separated by this quantity. If the posteriors become uniform, every code is used equally, the latent carries no preference, and perplexity attains its maximum of C. If instead the posteriors concentrate on a single code or a small subset, perplexity drops toward 1 and remains there. Reported together, the two quantities identify both whether the representation has collapsed and in which direction: mutual information near zero indicates collapse, and perplexity indicates whether the latent has spread uniformly or concentrated on a few codes.

Computation. Using the same per-observation probabilities $p_n^{(k)}$ as above, code perplexity is the exponential of the entropy of their average,

$$\mathrm{PP}\big(Z_k\big) \;=\; \exp\left(H\left(\frac{1}{N}\sum_{n=1}^{N} p_n^{(k)}\right)\right) \;\in\; [1, C], \tag{10}$$

which we average over the K latents. The exponential places the value on the scale of a number of codes: a latent whose average distribution is uniform over exactly m codes has perplexity m.

B Action Sensitivity experiment

Evaluating Action Sensitivity. Although applying SIGReg per distribution without a stop-gradient prevents latent collapse, downstream agents still fail to learn effective policies (Figure 4). To investigate this behavior, we examine how latent rollouts respond to action conditioning. For each trained model on Atari-5-100k, we initiate 16-step autoregressive rollouts conditioned on 8-step context sequences. At each step, we evaluate all available discrete actions and measure the mean pairwise Total Variation (TV) distance across the resulting categorical distributions, defined for two distributions p and q as $\mathrm{TV}(p,q) = \frac{1}{2}\sum_{c=1}^{C} |p_c - q_c|$, capturing the degree to which distinct agent decisions yield diverging future latent states.

Empirical Findings. Across both regularizer formulations, the stop-gradient operator proves essential for maintaining responsive action conditioning (Figure 7). Disabling the stop-gradient causes action sensitivity to drop substantially, as the encoder minimizes dynamics error by ignoring action inputs. While per-distribution regularization preserves a non-zero baseline of sensitivity where

the collapsed flattened model drops to zero, both SG-off configurations fall well short of the state divergence achieved with stop-gradients enabled. This attenuated action responsiveness limits the world model in distinguishing alternative choices, hindering the policy from correlating actions with rewards.

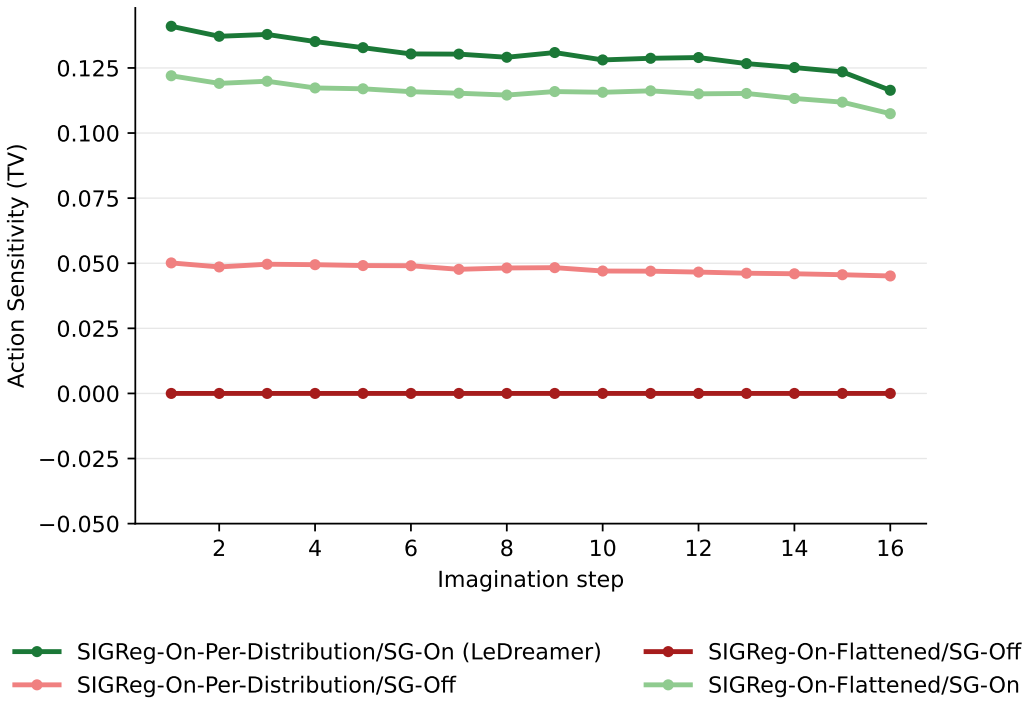

Figure 7: Action sensitivity measured via pairwise Total Variation (TV) across imagination rollout steps. The stop-gradient operator is essential for preserving action conditioning in the world model. Disabling stop-gradients severely attenuates this responsiveness, preventing actions from meaningfully steering imagined trajectories.

C Actor-Critic agent

The agent learns a policy purely from trajectories imagined by the world model, without any additional environment interaction. To initiate the imagination process, a short context trajectory is sampled from the replay buffer and encoded to obtain the initial posterior distributions $\mathcal{Z}_t$. We then sample $z_t \sim \mathcal{Z}_t$ and pass it through the action mixer together with a_t to obtain the tokens consumed by the sequence model, which returns the first prior logits. Beyond this context, no observations are available, so the latents are instead drawn from the predicted prior, $\hat{z}_t \sim \hat{\mathcal{Z}}_t$, and the world model rolls out autoregressively for a fixed imagination horizon L. Following Zhang et al. (2023), we employ a key-value cache to accelerate this inference. Predicted continuation flags are binarized during imagination rather than used as probabilities.

The agent state concatenates the current latent with the hidden state of the sequence model, which summarizes the interaction history:

$$s_t = [\, z_t,\, h_t \,]. \tag{11}$$

Actor–Critic Learning. We adopt the actor–critic formulation of DreamerV3 (Hafner et al., 2024), parameterized by θ and ψ respectively. We use the formulation prior to its 2024 update, in which the critic is trained with non-imagined data. The actor maximizes the discounted return, while the critic learns to approximate the distribution of returns under the current policy:

$$\begin{aligned} &\text{Actor:} && a_t \sim \pi_\theta(a_t \mid s_t), \\ &\text{Critic:} && V_\psi(s_t) \approx \mathbb{E}_{\pi_\theta, p_\phi}\Big[\textstyle\sum_{i=0}^{\infty} \gamma^i \hat{r}_{t+i}\Big]. \end{aligned} \tag{12}$$

Targets are bootstrapped λ-returns (Sutton & Barto, 2018) computed over the imagined trajectory from the rewards $\hat{r}_t$ and continuation flags $\hat{c}_t$ predicted by the world model, defined recursively backwards from the horizon L:

$$G_t^\lambda = \begin{cases} \hat{r}_t + \gamma \hat{c}_t \left[(1-\lambda) V_\psi(s_{t+1}) + \lambda G_{t+1}^\lambda\right], & t < L, \\ V_\psi(s_L), & t = L. \end{cases} \tag{13}$$

Following Hafner et al. (2024), the critic predicts a discretized distribution over returns rather than a scalar, and is trained with the symlog two-hot loss $\mathcal{L}^{\text{sym}}$. Following the official implementation of STORM, a second term regularizes it toward an exponential moving average of its own parameters, which stabilizes training without requiring a separate target network. In our implementation this target is the λ-return $G_t^{\lambda,\text{EMA}}$ recomputed with the averaged critic, rather than its value prediction at s_t:

$$\mathcal{L}(\psi) = \frac{1}{B_2 L} \sum_{n=1}^{B_2} \sum_{t=1}^{L} \Big[\mathcal{L}^{\text{sym}}\big(V_\psi(s_t), \text{sg}(G_t^\lambda)\big) + \mathcal{L}^{\text{sym}}\big(V_\psi(s_t), \text{sg}(G_t^{\lambda,\text{EMA}})\big)\Big], \tag{14}$$

where B_2 denotes the imagination batch size. The actor is trained with the REINFORCE estimator (Williams, 1992), using the critic as a baseline and an entropy bonus weighted by η to maintain exploration:

$$\mathcal{L}(\theta) = \frac{1}{B_2 L} \sum_{n=1}^{B_2} \sum_{t=1}^{L} \left[-\,\text{sg}\left(\frac{G_t^\lambda - V_\psi(s_t)}{\max(1, S)}\right) \ln \pi_\theta(a_t \mid s_t) - \eta\, \text{H}\big(\pi_\theta(a_t \mid s_t)\big)\right]. \tag{15}$$

Dividing the advantage by $\max(1, S)$ keeps the policy gradient at a comparable scale across environments whose reward magnitudes differ by orders of magnitude. We take S as the spread between the 5-th and 95-th percentiles of the λ-returns, and smooth each percentile over training with decay σ_S so that the normalizer does not fluctuate between consecutive batches. The critic parameter average uses decay σ_V:

$$\begin{aligned} \bar{P}_q &\leftarrow \sigma_S\, \bar{P}_q + (1-\sigma_S)\, P_q\big(G^\lambda\big), \qquad S = \bar{P}_{95} - \bar{P}_5, \\ \psi^{\text{EMA}} &\leftarrow \sigma_V\, \psi^{\text{EMA}} + (1-\sigma_V)\, \psi, \end{aligned} \tag{16}$$

where $P_q(\cdot)$ denotes the q-th percentile over the batch, L the imagination horizon, and $\text{H}(\cdot)$ the entropy of the policy distribution.

D ATARI-100K RESULTS

Evaluation Protocol and Aggregate Metrics To evaluate performance across diverse games, raw game scores are converted into Human-Normalized Scores (HNS). For a given game g, the normalized score is defined as:

$$\mathrm{HNS}_g = \frac{\mathrm{Score}_g - \mathrm{Random}_g}{\mathrm{Human}_g - \mathrm{Random}_g}, \tag{17}$$

where Random_g and Human_g are standard reference scores from the literature. The normalized mean and median are computed directly over per-game averages. Following the recommendations of Agarwal et al. (2022), we also compute the Interquartile Mean (IQM), which trims the top and bottom 25% of scores. The IQM is calculated across individual runs rather than per-game averages, and is marked as NA for baselines where individual seed trajectories are not publicly available.

To ensure a strictly fair comparison, all aggregate statistics are recomputed directly from each method's published per-game scores using the exact same reference baselines. This protocol reproduces the reported mean and median exactly for STORM, DreamerV3, DramaXS, and SGF. Two deviations remain: for EfficientZero-V2, we obtain a higher mean (2.680 vs. 2.428) and a slightly lower median (1.235 vs. 1.286), which may reflect different reference scores, and for MuDreamer, we obtain a lower median (0.337 vs. 0.430), although its mean closely matches the reported value (1.264 vs. 1.260). Since the corresponding evaluation code is not publicly available, we cannot determine the origin of these differences and report our recomputed values throughout.

Table 1: Comparison with other methods on the Atari 100k benchmark. Scores for LEDREAMER are averaged over 5 seeds, each evaluated on 100 episodes at the final checkpoint. Bold indicates the highest score, or within 5% of it, among methods that learn in imagination without lookahead search. The normalized mean and median are computed over per-game human-normalized scores. The interquartile mean is computed over individual runs rather than per-game averages, following Agarwal et al. (2022), and is reported as NA for methods whose per-run scores are not published.

			Lookahead	Learning in imagination					
				Decoder-enabled			Decoder-free		
Game	Random	Human	EZ-V2	DreamerV3	DramaXS	STORM	MuDreamer	SGF	LEDREAMER (std. dev.)
Alien	227.8	7127.7	1557.7	**959**	820	**984**	**951**	518.8	658.0 (128.7)
Amidar	5.8	1719.5	184.9	139	131	**205**	153	62.7	112.8 (29.3)
Assault	222.4	742.0	1757.5	706	539	801	**891**	**850.1**	731.7 (52.1)
Asterix	210.0	8503.3	61810.0	932	**1632**	1028	1411	802.5	681.9 (192.8)
Bank Heist	14.2	753.1	1316.7	**649**	137	**641**	156	58.7	371.1 (118.1)
Battle Zone	2360.0	37187.5	14433.3	12250	10860	**13540**	12080	3747.0	5122.0 (2190.4)
Boxing	0.1	12.1	75.0	78	78	80	**96**	83.4	41.2 (16.0)
Breakout	1.7	30.5	400.1	31	7	16	34	**50.7**	19.5 (2.5)
Chopper Command	811.0	7387.8	1196.6	420	1642	**1888**	808	1775.4	1404.4 (127.0)
Crazy Climber	10780.5	35829.4	112363.3	**97190**	83931	66776	**96128**	15751.3	59915.8 (5152.2)
Demon Attack	152.1	1971.0	22773.5	303	201	165	553	**2809.5**	204.3 (33.3)
Freeway	0.0	29.6	0.0	0	15	**34**	5	11.9	**33.1 (0.8)**
Frostbite	65.2	4334.7	1136.3	909	785	1316	**1652**	265.6	267.8 (7.3)
Gopher	257.6	2412.5	3868.7	3730	2757	**8240**	1500	416.4	868.0 (136.7)
Hero	1027.0	30826.4	9705.0	**11161**	7946	**11044**	8272	1522.9	9197.8 (1455.6)
James Bond	29.0	302.8	468.3	445	372	**509**	409	280.9	354.1 (52.0)
Kangaroo	52.0	3035.0	1886.7	4098	1384	**4208**	**4380**	271.2	2405.6 (1672.2)
Krull	1598.0	2665.5	9080.0	7782	**9693**	8413	**9644**	7813.7	6391.1 (705.2)
Kung Fu Master	258.5	22736.3	28883.3	21420	23920	**26182**	**26832**	20169.8	**27043.8 (3709.6)**
Ms Pacman	307.3	6951.6	2251.0	1327	2270	**2673**	2311	1356.8	1264.4 (170.4)
Pong	-20.7	14.6	20.8	**18**	15	11	**18**	12.6	5.1 (6.8)
Private Eye	24.9	69571.3	99.8	882	90	**7781**	1042	405.5	1002.6 (2024.5)
Qbert	163.9	13455.0	16058.3	3405	796	**4522**	4061	685.0	957.6 (203.5)
Road Runner	11.5	7845.0	27516.7	15565	14020	**17564**	8460	8164.2	12304.4 (3964.8)
Seaquest	68.4	42054.7	1974.0	**618**	497	525	428	476.8	286.2 (122.2)
Up n' Down	533.4	11693.2	15224.3	7667	7387	7985	**26494**	7745.0	2840.3 (1630.5)
Interquartile mean	0.000	1.000	NA	0.497	0.343	**0.636**	NA	NA	0.396
Normalized mean	0.000	1.000	2.680	1.124	1.049	**1.267**	**1.264**	0.884	0.768
Normalized median	0.000	1.000	1.235	0.490	0.270	**0.582**	0.337	0.152	0.279

E ATARI-5-100K: ABLATION BENCHMARK

Atari-5 Methodology. Aitchison et al. (2022) introduced a principled framework to compress the 57-game Arcade Learning Environment (ALE) (Bellemare et al., 2013) into a compact, representative subset. Their approach identifies the minimal collection of environments that best predicts aggregate

benchmark performance by fitting linear regression models over log-normalized game scores. To preserve weak ordering between algorithms and ensure that a random agent mapping to zero remains consistent, the regression is constrained to non-negative weights without an intercept. On the full ALE, this method yielded the 5-game subset Atari-5 (Battle Zone, Double Dunk, Name This Game, Phoenix, and Q*bert), which achieves an R^2 of 0.984 and estimates the full 57-game median within a 10.4% relative error. We refer readers to Aitchison et al. (2022) for the underlying mathematical proofs regarding order preservation.

Adaptation. The original Atari-5 benchmark cannot be applied to sample-efficient reinforcement learning. It was trained on high-data regime evaluations (50M to 200M frames), and three of its five games (Double Dunk, Name This Game, and Phoenix) are absent from the 26-game Atari 100k (Kaiser et al., 2024) suite. Although Atari 100k requires fewer environment interactions per run, evaluating full ablation sweeps across all 26 games and multiple seeds remains computationally expensive. Consequently, prior work frequently resorts to subjective, ad hoc selections of games to demonstrate ablations. Because these game choices are not standardized across different works, it is difficult to determine whether observed modifications actually translate to performance gains on the full benchmark.

Derivation of Atari-5-100k. To provide a representative and standardized testbed for ablations, we adapt the methodology of Aitchison et al. (2022) to the Atari 100k (Kaiser et al., 2024) setting. We collected published per-game average scores across 32 recent sample-efficient world models evaluated on the Atari 100k benchmark. Following the recommendations of Agarwal et al. (2022), we select the interquartile mean as our target summary metric. Our target differs from the estimator they define, which is computed across individual runs: the majority of the 32 methods in our dataset publish only per-game averages rather than per-seed results, so a run-level IQM cannot be computed for them. We therefore use the IQM over per-game means, and refer to it as such throughout to distinguish it from the run-level estimator. We prefer it to the mean or the median over the same per-game scores because it retains the property that motivates the IQM in the first place: discarding the top and bottom 25% of values leaves the target insensitive to the small number of games in which a method scores far above or far below the rest of the suite, while still aggregating over the remaining half of the distribution rather than a single point. Following the original subset search formulation, we fit intercept-free linear regression models with non-negative weights over log-normalized scores, and select the five-game subset minimizing leave-one-out cross-validated error against this target. We exclude Freeway from the candidate pool, as training and evaluation protocols for this environment vary substantially across methods: STORM supplies an initial demonstration trajectory, while IRIS employs a game-specific policy temperature. Freeway is retained in the target metric, so the subset predicts an aggregate computed over all 26 games. The resulting subset (Bank Heist, Breakout, James Bond, Kangaroo, and Kung-Fu Master) constitutes the Atari-5-100k ablation benchmark used throughout our experimental analyses. It attains an in-sample R^2 of 0.946, a leave-one-out cross-validated R^2 of 0.922, and a cross-validated relative prediction error of 2.68%, compared to the in-sample 0.984 and 10.4% reported for Atari-5.

Correlation between games. In their analysis of the full Arcade Learning Environment, Aitchison et al. (2022) observed that many games are highly correlated with one another. For this reason, we present the correlation matrix for our selected Atari-5-100k subset in Figure 8. In general, the games within our subset are only weakly correlated, with the strongest pair at 0.52. This lack of correlation suggests that performance across these environments tests distinct, independent agent capabilities, ensuring a diverse and comprehensive evaluation.

Dataset We conducted an extensive literature search to compile a comprehensive collection of sample-efficient world models with published results on the Atari 100k benchmark. Table 2 details the 32 models identified through this search.

F Ablations on architecture parameters

All ablation studies were conducted on our proposed Atari-5-100k benchmark. Unless otherwise specified, the reported reward curves represent an average across three independent training seeds, evaluated over 20 episodes every 20,000 environment steps. For the evaluation of our primary

Table 2: List of methods.

#	Method
1	REM Cohen et al. (2024)
2	DreamerV3 Hafner et al. (2024)
3	Vector-Cutie-OC-DreamerV3 Zhang et al. (2026)
4	EADream Peng et al. (2026)
5	TWM Robine et al. (2023)
6	HarmonyDreamerV3 Ma et al. (2024)
7	IRIS Micheli et al. (2023)
8	PO-Dreamer Li et al. (2026)
9	TWISTER Burchi & Timofte (2025)
10	DART Agarwal et al. (2024)
11	GIT-STORM Meo et al. (2025)
12	JEDI32 Lim et al. (2026)
13	STORM Zhang et al. (2023)
14	Vector-Cutie-OC-STORM Zhang et al. (2026)
15	DIAMOND Alonso et al. (2024)
16	Mask-Focus-STORM Zhang et al. (2026)
17	SimPLe Kaiser et al. (2024)
18	SGF Robine et al. (2025)
19	DELTA-IRIS Micheli et al. (2024)
20	Simulus Cohen et al. (2026)
21	JEDI64 Lim et al. (2026)
22	DyMoDreamer Zhang et al. (2025)
23	DramaXS Wang et al. (2025)
24	DreamerV3XS Wang et al. (2025)
25	EASimulus Peng et al. (2026)
26	EMERALD Burchi & Timofte (2026)
27	DramaXS-uniform Wang et al. (2025)
28	EDELINE Lee et al. (2025)
29	Hieros Mattes et al. (2024)
30	ITC Kim et al. (2026)
31	Vector-SAM2-OC-STORM Zhang et al. (2026)
32	MuDreamer Burchi & Timofte (2024)

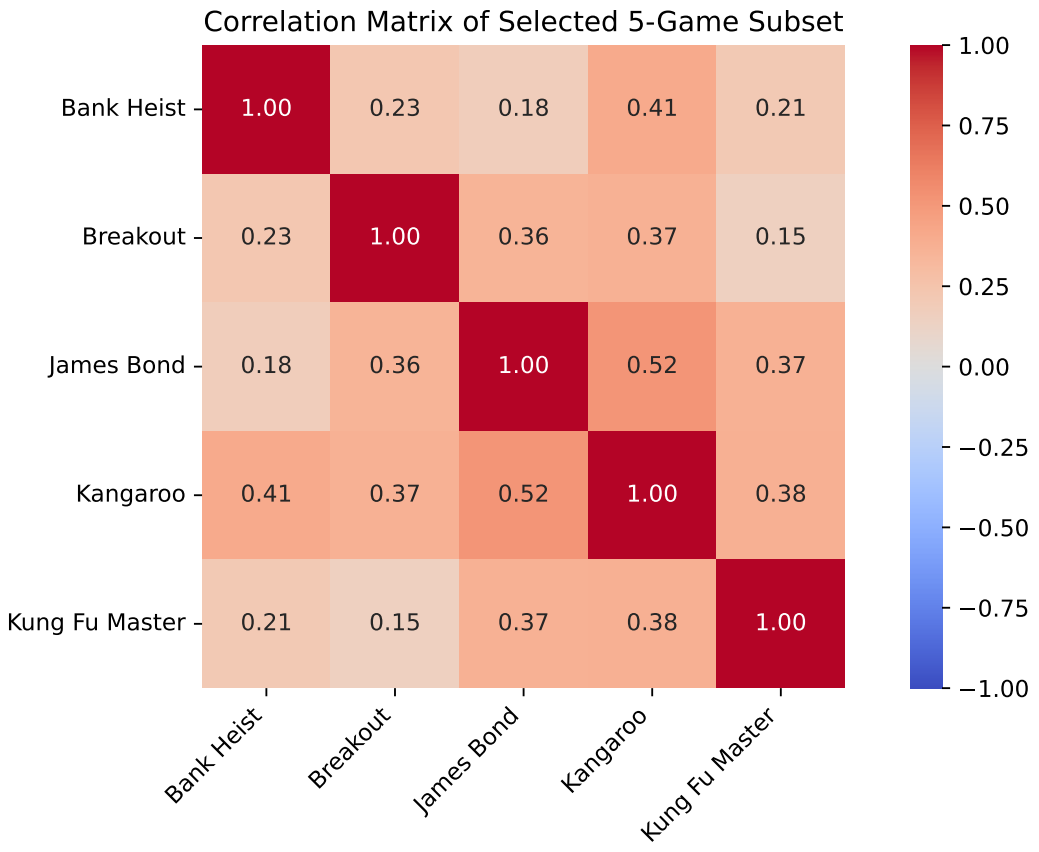

Figure 8: Pearson correlation matrix of the five environments comprising the Atari-5-100k ablation benchmark. The correlation coefficients are calculated using the log-transformed Human-Normalized Scores (HNS) achieved by the 32 evaluated sample-efficient world models.

model, LeDreamer, we expand this to an average across five independent training seeds. While we maintained the standard 20 evaluation episodes at 20,000-step intervals throughout training, the final policy at 100,000 environment steps was comprehensively evaluated over 100 episodes.

Number of Distributions per Latent: Figure 9 illustrates the impact of varying the number of categorical distributions within the latent state on agent performance. While increasing the number of distributions generally improves performance, the benefits converge relatively quickly. The highest overall agent performance was recorded at 16 distributions.

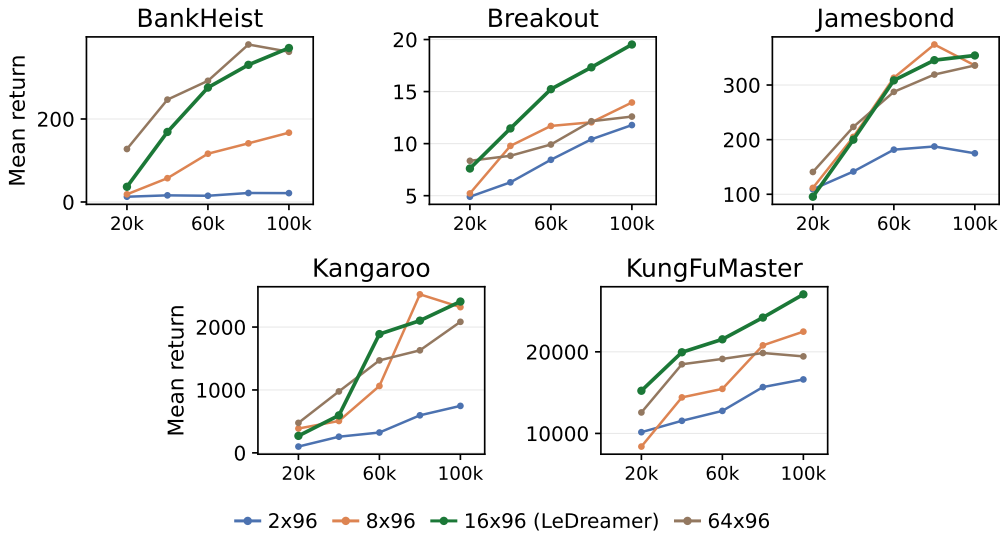

Figure 9: Mean episodic returns evaluated on the Atari-5-100k benchmark across varying numbers of categorical distributions per latent.

Number of Codes per Distribution: Figure 10 evaluates the effect of the number of codes per distribution. In contrast to the findings in LeWM (Maes et al., 2026), agent performance remains robust at low dimensionalities. Specifically, utilizing only 8 codes per distribution remains competitive, with the highest mean return on Kangaroo, although this game exhibits high variance across seeds. This indicates that a more computationally efficient variant of LeDreamer could effectively leverage a reduced code count. We ultimately selected 96 codes to align with the methodology of LeWM, which demonstrated performance saturation between 96 and 384. For applications requiring further reduction, a configuration of 32 codes achieves results on par with the 96-code baseline.

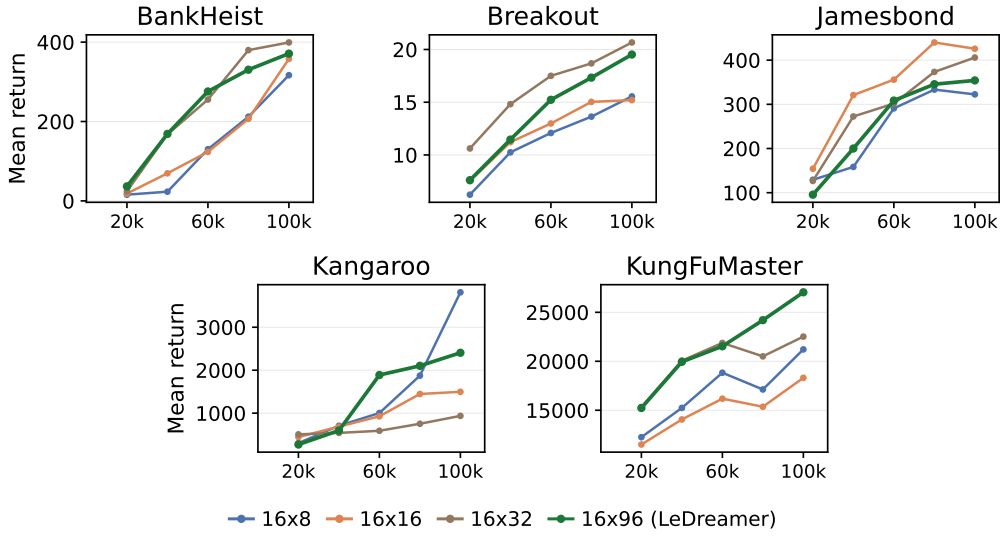

Figure 10: Mean episodic returns evaluated on the Atari-5-100k benchmark across varying numbers of codes per categorical distribution.

Update Frequency (Gradient Steps per Environment Step): Figure 11 details the performance trade-offs associated with altering the ratio of gradient updates to environment interactions. To explore accelerated training configurations for LeDreamer, we reduced the update frequency. Executing one training step every two environment steps halves the total training time while only minimally affecting final performance. However, decreasing the frequency beyond this threshold results in a substantial deterioration of the learned policy.

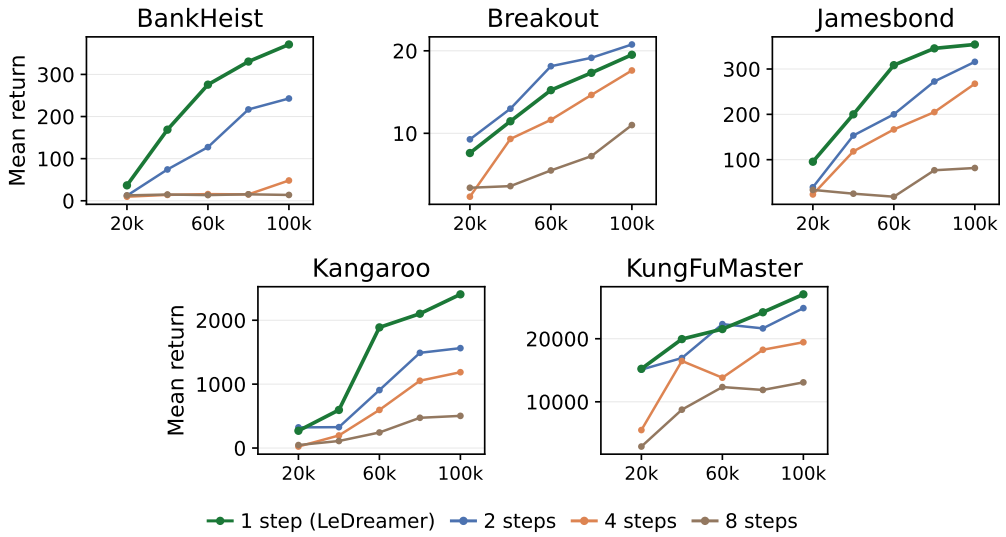

Figure 11: Mean episodic returns evaluated on the Atari-5-100k benchmark across varying update frequencies (gradient steps per environment interaction).

Dynamics Objective (MSE vs. KL-Balancing): When applying the standard KL-balancing loss formulation used in STORM (Zhang et al., 2023), the model fails to learn representations because the loss magnitude consistently remains below the free bits threshold of 1, resulting in zero gradient updates. To ensure a fair evaluation, we removed the free bits threshold, allowing the model to train. However, standard KL-balancing still produced highly uncertain and unstable dynamics, which severely hindered policy optimization. To further investigate this, we also trained the model using a dynamics-only KL divergence objective by removing the representation loss term, which yielded noticeably improved performance. We attribute this difference to the representation loss: because it lacks a stop-gradient on the posterior target, it constantly updates the encoder to make next-step prediction trivially easy, destabilizing representations under non-stationary online data distributions. Figure 12 presents the reward curves comparing these objectives, where our MSE formulation obtains the highest mean returns.

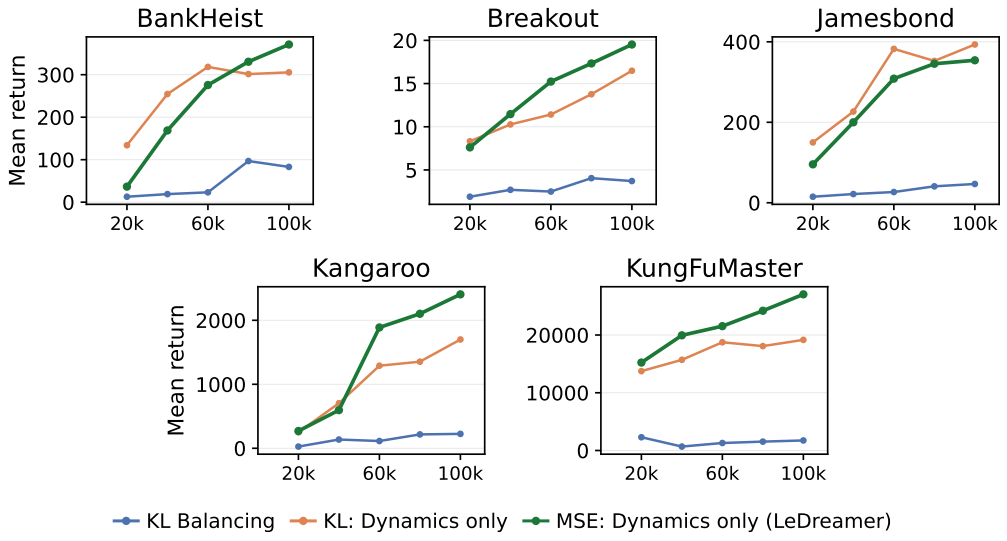

Figure 12: Mean episodic returns on the Atari-5-100k benchmark across different dynamics loss formulations: Mean Squared Error on log probabilities, standard KL-balancing, and dynamics-only KL divergence (without representation loss).

Batch Size in Per-Distribution Regularization: LeWM (Maes et al., 2026) utilizes a significantly larger batch size (128) compared to our baseline (32). This distinction is highly relevant because a larger batch size increases the number of unit projections considered, which theoretically enhances SIGReg performance. Figure 13 illustrates the impact of batch size variations on our architecture. We observe that agent performance remains stable and largely independent of the batch size within our framework.

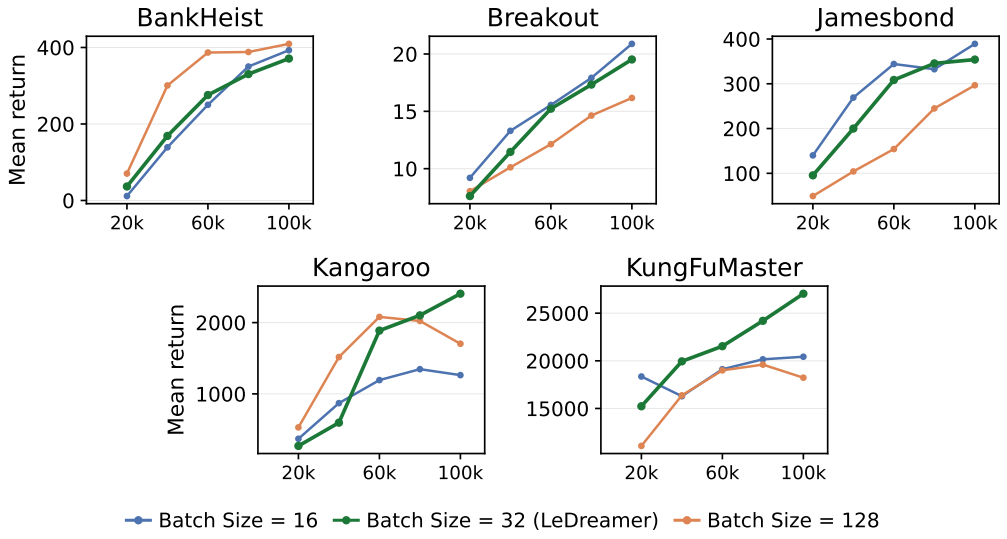

Figure 13: Mean episodic returns on the Atari-5-100k benchmark across different batch sizes.

Batch Size in Flattened Regularization: A potential concern with the flattened baseline is that regularizing a 1,536-dimensional latent space with a small batch size ($B = 32$) might suffer from sample starvation during characteristic function estimation. To test whether increasing the sample

budget resolves this limitation, we evaluated the flattened configuration using a four-fold larger batch size ($B = 128$), matching the setup in LeWM (Maes et al., 2026). As shown in Figure 14, increasing the batch size fails to close the performance gap. Per-distribution regularization with $B = 32$ obtains higher mean returns on most games compared to the flattened model with $B = 128$. This suggests that the advantage of our approach stems from the structural inductive bias of constraining each categorical distribution independently, rather than from an artifact of sample size during projection estimation.

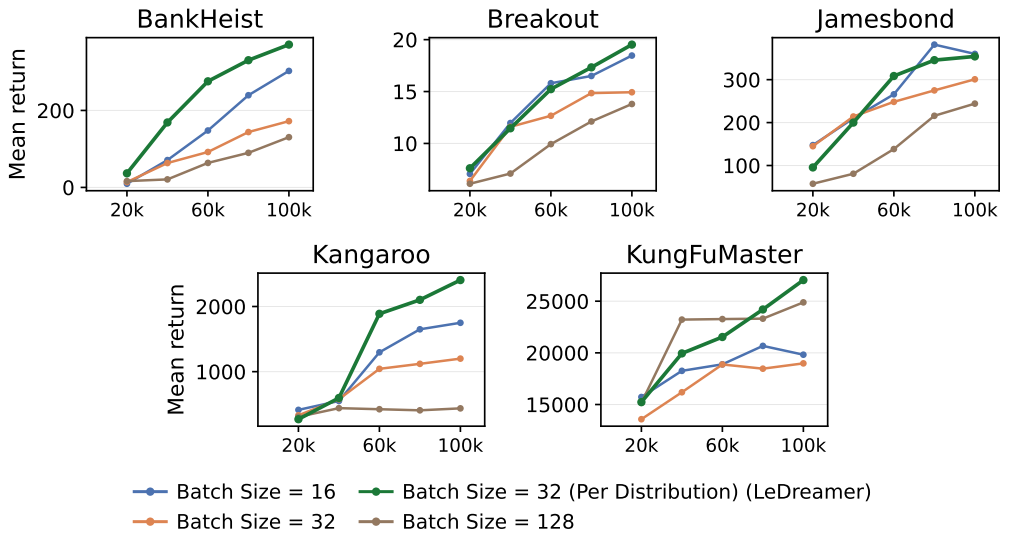

Figure 14: Mean episodic returns on the Atari-5-100k benchmark across different batch sizes with the flattened configuration. LeDreamer, which uses Per-Distribution regularization, is included as a reference point of performance.

Alternative Regularization Methods: Although we adopted SIGReg for its theoretical property of targeting the optimal distribution to minimize downstream risk, we investigated whether the benefits of per-distribution regularization extend to other anti-collapse mechanisms. Specifically, we adapted the VICReg objective from SGF (Robine et al., 2025) and the batch normalization approach from MuDreamer (Burchi & Timofte, 2024) to operate on our encoder logits across both flattened and per-distribution configurations. As shown in Figure 15, the advantage of per-distribution constraints also benefits VICReg, which obtains higher mean returns than its flattened counterpart, though this distinction is less pronounced for batch normalization. Furthermore, we observe that even with the stop-gradient operator enabled, applying batch normalization to the encoder logits obtains markedly lower mean returns than SIGReg.

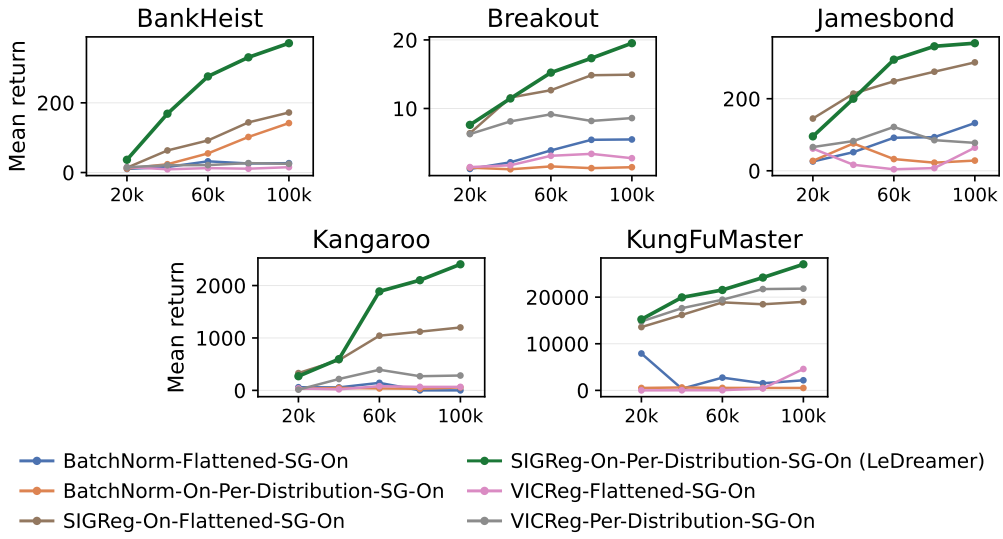

Figure 15: Mean episodic returns on the Atari-5-100k benchmark across different representation anti-collapse methods.

Number of Projections (SIGReg): Consistent with the findings in LeWM (Maes et al., 2026), agent performance remains largely invariant across different numbers of random unit projections M in SIGReg. Increasing the projection count does not provide noticeable performance gains, as $M = 512$ generally achieves higher returns than $M = 1024$. Furthermore, a substantially smaller projection budget such as $M = 64$ remains competitive with $M = 512$, obtaining the top score on James Bond and the second-highest return on Breakout (Figure 16).

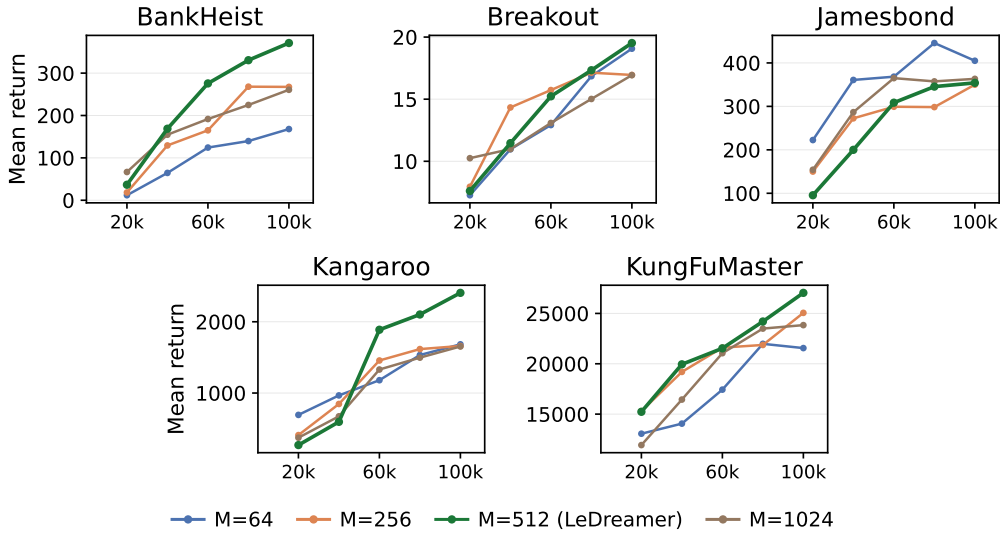

Figure 16: Mean episodic returns evaluated on the Atari-5-100k benchmark across varying numbers of random unit projections M.

Regularization Weight (λ_{SIGReg}): We evaluated sensitivity to the regularization coefficient λ_{SIGReg} across both the per-distribution and flattened configurations. As shown in Figure 17, performance remains relatively stable across varying values of λ_{SIGReg}, with no single setting

uniformly outperforming the others across all environments. In LeWM (Maes et al., 2026), larger coefficients such as $\lambda_{\text{SIGReg}} = 0.5$ caused a noticeable degradation in returns. In our setting, however, this pattern is less definitive: $\lambda_{\text{SIGReg}} = 0.5$ achieves strong performance on Kangaroo and Kung-Fu Master, while showing lower returns on Bank Heist, Breakout, and James Bond. These findings suggest that the choice of regularization weight is not a crucial hyperparameter in our framework.

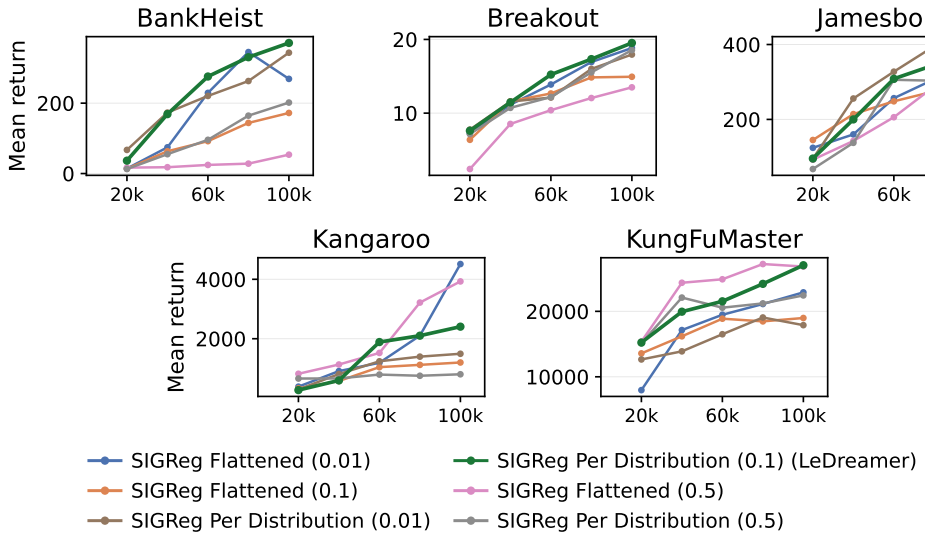

Figure 17: Mean episodic returns evaluated on the Atari-5-100k benchmark across varying values of λ_{SIGReg}.

G Computational Cost

Most of the development of LeDreamer was carried out on a single RTX 5080, and the majority of training runs were performed on that same GPU together with an RTX 3070. For the final results and ablations, these two GPUs were complemented by an RTX 4090 and two Quadro RTX 8000. A single training run of LeDreamer, on one environment with a single seed, takes 1 hour and 33 minutes on an RTX 5080.

Robine et al. (2025) follow Hafner et al. (2024) in estimating runtimes as NVIDIA V100 GPU hours, assuming that a P100 is twice as slow and an A100 twice as fast. We instead train all models on the same machine, equipped with an RTX 5080. However, we could not train MuDreamer from scratch, as its code has not been released, we therefore take its reported runtime of 4 hours and 40 minutes on an RTX 3090, compare it against the runtime reported for STORM on the same GPU, 4.3 hours, and use this ratio to approximate its runtime in RTX 5080 hours. EfficientZero-V2 is approximated in the same way from its reported 2.7 hours on 8 RTX 3090 GPUs, scaled to a single GPU under the assumption of ideal parallel scaling, which makes the resulting estimate an upper bound.

H DETAILS OF ARCHITECTURE

Table 3: Structure of the encoder. Each convolutional layer uses kernel $= 4$, stride $= 2$ and padding $= 1$, and is applied without bias. GN denotes a group normalization layer (Wu & He, 2018) with a single group, and SiLU (Elfwing et al., 2017) refers to the sigmoid linear unit used for activation.

Submodule	Output tensor shape
o_t	$3 \times 64 \times 64$
Conv1 + GN1 + SiLU	$32 \times 32 \times 32$
Conv2 + GN2 + SiLU	$64 \times 16 \times 16$
Conv3 + GN3 + SiLU	$128 \times 8 \times 8$
Conv4 + GN4 + SiLU	$256 \times 4 \times 4$
Flatten	4096
Linear	$K \cdot C$
Reshape (produce Z_t)	$K \times C$

Table 4: Structure of the action mixer. As a first step, z_t and a_t are combined by concatenating them along the last dimension. The positional encoding is a learnable embedding matrix of shape $T \times D$, where T is the sequence length, and is added to the output of the mixer.

Submodule	Output tensor shape
z_t, a_t	$K \times C$, A
Reshape and concatenate	$K \cdot C + A$
Linear1 + LN1 + SiLU	D
Linear2 + LN2	D
Add positional encoding (produce e_t)	D

Table 5: Transformer block. The block follows a post-normalization arrangement, in which the residual connection is applied before each layer normalization.

Submodule	Output tensor shape
Input features (x_1)	$T \times D$
Multi-head self attention + Dropout(p)	$T \times D$
Residual (x_1)	$T \times D$
LN1 (x_2)	$T \times D$
Linear1 + SiLU + Dropout(p)	$T \times 2D$
Linear2 + Dropout(p)	$T \times D$
Residual (x_2)	$T \times D$
LN2	$T \times D$

Table 6: Transformer based sequence model. The positional encoding is applied inside the action mixer (Table 4) and the Transformer block is explained in Table 5. A causal mask is applied to the self-attention so that e_t attends only to $e_1, e_2, \ldots, e_t$. During imagination, a key-value cache is used to accelerate inference.

Submodule	Output tensor shape
Input ($e_{1:T}$)	
Transformer blocks $\times 2$	$T \times D$
Output ($h_{1:T}$)	

Table 7: MLP structures. Every hidden layer is applied without bias and is followed by layer normalization (Ba et al., 2016) and a SiLU (Elfwing et al., 2017) activation. 255 is the size of the bucket of the symlog two-hot loss (Hafner et al., 2024), and A denotes the action dimension, which ranges from 3 to 18 across different games. The agent state $s_t = [z_t, h_t]$ has dimension $K \cdot C + D$.

Module name	Symbol	MLP layers	Input/ MLP hidden/ Output dimension
Dynamics head	g_ϕ^D	1	D/ -/ $K \cdot C$
Reward predictor	g_ϕ^R	3	D/ D/ 255
Continuation predictor	g_ϕ^C	3	D/ D/ 1
Policy network	$\pi_\theta(a_t \mid s_t)$	3	$K \cdot C + D$/ D_{agent}/ A
Critic network	$V_\psi(s_t)$	3	$K \cdot C + D$/ D_{agent}/ 255

I HYPERPARAMETERS

Table 8 summarizes the complete set of hyperparameters used across our framework. Our Transformer world model architecture largely adheres to the configuration of STORM (Zhang et al., 2023), with the exception of dropout, which we omit as we observed no empirical evidence of overfitting. Similarly, our actor-critic agent adopts the hyperparameter settings of STORM, with a relevant distinction: for the game Freeway, rather than utilizing an initial demonstration trajectory, we follow the evaluation protocol of IRIS (Micheli et al., 2023) and employ a lower policy sampling temperature during evaluation.

Table 8: Hyperparameters

Hyperparameter	Symbol	Value
World Model batch size	B_1	32
World Model sequence length	T	24
Distributions per latent	K	16
Codes per distribution	C	96
Agent batch size	B_2	512
Agent context length	-	8
Agent imagination horizon	L	16
Train steps every env step	-	1
Environment context length	-	16
World Model Optimizer	-	Adam
World Model learning rate	-	0.0001
World Model gradient clipping	-	1.0
Transformer layers	-	2
Transformer feature dimension	D	512
Transformer heads	-	8
Dropout probability	p	0.0
Stop-gradient on posterior target	-	True
MSE dynamics loss	-	True
SIGReg weight	λ_{sigreg}	0.1
SIGReg number of projections	M	512
SIGReg integration knots	-	17
SIGReg per distribution	-	True
Agent Optimizer	-	Adam
Agent learning rate	-	0.00003
Agent gradient clipping	-	100.0
Agent hidden dimension	D_{agent}	1024
Gamma	γ	0.985
Lambda	λ	0.95
Entropy coefficient	η	0.0003
Critic EMA decay	σ_V	0.98
Percentile EMA decay	σ_S	0.99
Random actions during collection	-	0.1
Policy temperature for evaluation	-	1.0 (0.01 for Freeway)
Gray scale input	-	False
Frame stacking	-	False
Frame skipping	-	4 (max over last 2 frames)
Use of life information	-	True

J WHY SIGREG MUST BE APPLIED PER CATEGORICAL DISTRIBUTION: A FEE-VERSUS-WALL ANALYSIS

This appendix isolates the theory behind the central design choice of LeDreamer—sketched isotropic regularization applied to each categorical distribution independently—in three self-contained steps. First, an *invariance*: the softmax reads only within-block differences of logits, so every task loss is blind to a K-dimensional gauge subspace, and the manifold of totally collapsed, zero-information representations is reachable at exactly zero task cost (Lemma 1). Second, a *projection law*: a sketched isotropy test prices a trace-matched degeneracy by the dimension of its support, at rate $1/r$, not by the ambient dimension (Lemma 2). Third, a *two-scale argument*: no single scalar law can approximate the fixed target $\mathcal{N}(0,1)$ at two separated scales at once. Combining them (Theorem 3): the flattened regularizer prices total collapse at a *fee* of order $1/K$, which the optimizer pays out of its dynamics savings, whereas the per-distribution regularizer erects a *wall*—a constant penalty, uniform over everything the encoder controls. Proposition 5 shows the converse direction: blocking has no intrinsic power, and the per-distribution test works because it is sliced exactly along the invariance that the softmax fixes. The design principle: *sketched-isotropy regularization must be applied at the level at which the architecture defines its invariances*—here, one test per categorical distribution.

J.1 SETTING AND ASSUMPTIONS

We analyze the population idealization of the two regularizer variants in equation 6. Fix a (timestep, latent) group and let $\ell \in \mathbb{R}^d$, $d = KC$ (with $C \geq 2$), denote the concatenated pre-softmax logits of one observation, viewed as a random vector under the data distribution; its block $\ell^{(k)} \in \mathbb{R}^C$ parameterizes the k-th categorical distribution $Z_k = \mathrm{softmax}(\ell^{(k)})$, from which the code is sampled. We write P for the law of ℓ.

Gauge decomposition. Let $b_k \in \mathbb{R}^d$ be the unit vector supported on block k with entries $1/\sqrt{C}$, so that $\{b_1, \ldots, b_K\}$ is orthonormal. That is,

$$b_k \;:=\; \tfrac{1}{\sqrt{C}}\big(0,\ldots,0,\underbrace{1,\ldots,1}_{\text{block } k},0,\ldots,0\big) \;\in\; \mathbb{R}^d.$$

Define the gauge subspace $G := \mathrm{span}\{b_1, \ldots, b_K\}$, $\dim G = K$, with orthogonal projector Π_G, and decompose

$$\ell \;=\; \sum_{k=1}^{K} \mu_k\, b_k \;+\; \eta, \qquad \mu_k := \langle \ell, b_k \rangle = \tfrac{1}{\sqrt{C}} \sum_c \ell_c^{(k)}, \qquad \eta := \ell - \Pi_G \ell, \tag{18}$$

so that blockwise $\eta^{(k)} = \ell^{(k)} - \big(\frac{1}{C}\sum_c \ell_c^{(k)}\big)\mathbf{1}_C$ are the centered logits, $\eta^{(k)} \perp \mathbf{1}_C$.

Statistic. Let ρ be a finite nonnegative measure on $\mathbb{R}$ (the test weight). For a real random variable X with characteristic function φ_X, define the population Epps–Pulley-type statistic against the fixed standard normal target,

$$T_\rho(X) \;:=\; \int_{\mathbb{R}} \big|\varphi_X(t) - e^{-t^2/2}\big|^2 \,\mathrm{d}\rho(t). \tag{19}$$

The two regularizer variants (Section 2), in population form—restating equation 7 of the main text—are

$$S_\rho^{\mathrm{flat}}(P) := \mathbb{E}_{u\sim \mathrm{Unif}(\mathbb{S}^{d-1})}\, T_\rho\big(\langle u, \ell\rangle\big), \qquad S_\rho^{\mathrm{pd}}(P) := \frac{1}{K}\sum_{k=1}^{K} \mathbb{E}_{v\sim \mathrm{Unif}(\mathbb{S}^{C-1})}\, T_\rho\big(\langle v, \ell^{(k)}\rangle\big). \tag{20}$$

Throughout, $\bar\kappa_\rho := \frac{1}{4}\int t^4 \,\mathrm{d}\rho(t)$; for the standard Gaussian weight $\mathrm{d}\rho = (2\pi)^{-1/2} e^{-t^2/2}\mathrm{d}t$ one has $\bar\kappa_\rho = 3/4$, and the sharp local curvature constant $\kappa_\rho := \frac{1}{4}\int t^4 e^{-t^2}\,\mathrm{d}\rho(t) = \frac{1}{12\sqrt{3}} \approx 0.048$ appears only in Remark 8.

All statements below concern the population quantities equation 20, i.e., the limit of infinitely many projection directions and batch elements. This is the strong form of the negative result: the fee of Theorem 3(a) is a population bias, not an estimation artifact, and cannot be removed by increasing the batch size B_1 or the number of projections M (cf. Figure 14); finite-sample deviations are standard and quantified in Remark 8.

Assumption 1 (Weight)**.** ρ is a finite nonnegative measure with $\int t^4 \, \mathrm{d}\rho < \infty$. For Theorem 3(b) we additionally assume that ρ has atoms at two points $0 < t_1 < t_2$, with masses $\rho_1, \rho_2 > 0$—as in our implementation, a quadrature with 17 knots. (Absolutely continuous weights, e.g. the Gaussian, yield the same conclusion; see Remark 3.)

Assumption 2 (Fixed target)**.** The test compares against the fixed law $\mathcal{N}(0, 1)$, with no per-projection standardization of the projected sample, as in LeJEPA (Balestriero & LeCun, 2025) and in our implementation. This is load-bearing: the wall of Theorem 3(b) detects scale mismatches, which standardization would erase.

Assumption 3 (No logit-scale regularization)**.** No training term outside Eq. 3 prices the scale of the encoder logits: no weight decay on the final encoder projection and no explicit logit normalization. (Our configuration uses Adam without weight decay, Table 8.) This is required for the exactness of the zero-task-cost claim in Lemma 1; under weight decay the gauge coordinates would be priced indirectly through parameter norms.

J.2 The gauge subspace of the softmax

Intuition. The softmax reads only differences of logits within a block: adding a constant to all C logits of one distribution changes nothing downstream—not the probabilities, not the sampled code, not the dynamics targets, not the reward or continuation predictions. The K per-block means are therefore a blind spot shared by every task term: K scalar channels, per observation, in which the encoder can place arbitrary input-dependent variance at exactly zero task cost. This also identifies the collapse manifold exactly: the representation carries no information precisely when the centered logits are constant across observations, and there the dynamics loss attains zero.

Lemma 1 (Gauge invariance and the zero-information manifold)**.** *Let $g : \mathcal{O} \to G$ be any measurable map and consider the replacement $\ell(o) \mapsto \ell(o) + g(o)$. Under Assumption 3:*

(i) Every term of the LeDreamer objective except the regularizer is unchanged. Equivalently, the task loss is a function of η only, and $\nabla_\mu L_{\text{task}} \equiv 0$; the implemented backward pass annihilates the gauge direction as well, since the softmax Jacobian $J = \mathrm{diag}(p) - pp^\top$ satisfies $J\mathbf{1}_C = 0$.

(ii) Define the zero-information manifold $\mathcal{C}_0 := \{P : \eta \text{ is } P\text{-a.s. constant}\}$. Then $P \in \mathcal{C}_0$ if and only if $I(O; Z_k) = 0$ for all k; on $\mathcal{C}_0$ the dynamics loss (Eq. 4) attains the value 0 (the target $\log Z_{t+1}$ is the same constant vector for every observation and action), and the imagined next-state distribution is independent of the action, so the action sensitivity measured in Appendix B is identically zero.

Proof. (i) For any $c \in \mathbb{R}$, $\mathrm{softmax}(\ell^{(k)} + c\mathbf{1}_C) = \mathrm{softmax}(\ell^{(k)})$ and $\log \mathrm{softmax}(\ell^{(k)} + c\mathbf{1}_C) = \log \mathrm{softmax}(\ell^{(k)})$, since numerator and denominator of the softmax are both multiplied by e^c and logsumexp shifts by c. Hence the posteriors Z_t, the sampled codes z_t, the straight-through estimator (whose forward pass uses z_t and whose backward pass uses J), the dynamics target and prediction (functions of probabilities only), and the reward/continuation heads (functions of z_t and actions) are all invariant. Every task term is therefore a function of the centered logits, and all partial derivatives in the μ-coordinates vanish.

(ii) If η is a.s. constant, all posteriors are a.s. equal to a fixed $p^{(k)}$, so Z_k is independent of O and $I(O; Z_k) = 0$; the dynamics target is the constant vector $\log p^{(k)}$, attained exactly by a constant-output predictor, and branching over actions leaves the (constant) prediction unchanged. Conversely, $I(O; Z_k) = H\big(\mathbb{E}_o[p^{(k)}(o)]\big) - \mathbb{E}_o\big[H(p^{(k)}(o))\big]$ (Eq. 9), which vanishes iff equality holds in Jensen's inequality for the strictly concave entropy, iff $p^{(k)}(o)$ is a.s. constant, iff $\eta^{(k)}(o)$ is a.s. constant (softmax is injective on the subspace orthogonal to $\mathbf{1}_C$). □

Remark 1 (Scope of the blind spot). Lemma 1 does not merely exhibit one bad point: the task loss is invariant under an infinite-dimensional group of deformations (K scalars per observation), so its minimizers come in gauge orbits. Reward and continuation terms are also gauge-invariant; collapse does degrade them through the uninformative codes, but these signals are sparse and empirically insufficient to maintain diversity on their own (the SIGReg-Off run of Figure 3 also collapses). The bookkeeping below therefore prices collapse against the dynamics term, the dominant representation-shaping signal.

J.3 What a sketched isotropy test certifies

Intuition. By the Cramér–Wold device, matching the fixed target $\mathcal{N}(0,1)$ along all directions characterizes $\mathcal{N}(0,I)$. The finite-power question is different: how hard does the test push against a specific violation? The answer is governed by the dimension of the violation's support, not by the ambient dimension. A trace-matched Gaussian of rank r casts one-dimensional shadows that are exactly Gaussian, with random variance concentrating around 1 with fluctuations of order $\sqrt{2/r}$ (Diaconis & Freedman, 1984): for moderate r, almost every direction looks almost perfect. At the other extreme, $r = 1$ casts χ^2_1-distributed shadow variances (median 0.455; a quarter of all directions see variance below 0.1): unmissable. The relevant parameter is the dimension of the hiding place, not of the vector.

Lemma 2 (Support-dimension law)**.** *Let Π be an orthogonal projector of rank r on $\mathbb{R}^n$ and let $X \sim \mathcal{N}\big(0, \frac{n}{r}\Pi\big)$ (trace-matched: $\operatorname{tr}\Sigma = n$). For $u \sim \mathrm{Unif}(\mathbb{S}^{n-1})$ the projection is exactly Gaussian, $\langle u, X\rangle \sim \mathcal{N}(0, V(u))$ with $V(u) = \frac{n}{r}\|\Pi u\|^2$, and*

$$\|\Pi u\|^2 \sim \mathrm{Beta}\big(\tfrac{r}{2}, \tfrac{n-r}{2}\big), \qquad \mathbb{E}[V] = 1, \qquad \mathrm{Var}(V) = \frac{2(n-r)}{r(n+2)} \;\le\; \frac{2}{r}. \tag{21}$$

Consequently, for any weight satisfying Assumption 1,

$$\mathbb{E}_u\, T_\rho\big(\langle u, X\rangle\big) \;\le\; \bar\kappa_\rho\, \mathbb{E}\big[(V-1)^2\big] \;=\; \bar\kappa_\rho\, \frac{2(n-r)}{r(n+2)} \;\le\; \frac{2\bar\kappa_\rho}{r}. \tag{22}$$

For $r = 1$, $V = n\langle u, e\rangle^2$ with $\langle u, e\rangle^2 \sim \mathrm{Beta}\big(\frac{1}{2}, \frac{n-1}{2}\big)$, and the law of V converges to χ^2_1 as $n \to \infty$: the shadow variances are $\Theta(1)$-dispersed.

Proof. Write $u = \xi/\|\xi\|$ with $\xi \sim \mathcal{N}(0, I_n)$; then $\|\Pi u\|^2 = A/(A+B)$ with $A \sim \chi^2_r$ independent of $B \sim \chi^2_{n-r}$, which is $\mathrm{Beta}(r/2, (n-r)/2)$. Its mean is r/n, giving $\mathbb{E}V = 1$, and its variance is $\frac{ab}{(a+b)^2(a+b+1)}$ with $a = r/2$, $b = (n-r)/2$, giving equation 21. For the transfer to the statistic, note that for $X \sim \mathcal{N}(0, V)$, $\varphi_X(t) = e^{-Vt^2/2}$, and by the mean value theorem, for all $V \ge 0$, $\big|e^{-Vt^2/2} - e^{-t^2/2}\big| \le \frac{t^2}{2}|V-1|$, since $\big|\partial_s e^{-st^2/2}\big| \le t^2/2$ for $s \ge 0$. Squaring and integrating against ρ, $T_\rho(\mathcal{N}(0,V)) \le \bar\kappa_\rho (V-1)^2$, and equation 22 follows by taking expectations, using $\mathbb{E}V = 1$. □

J.4 Main result: the fee and the wall

Intuition. The cheat prices itself. Set the centered logits to zero—uniform posteriors, zero information, zero dynamics loss—and let the K per-block means carry whitened features of the observation, with total variance matched to the isotropic target. To the flattened test, this configuration is exactly the trace-matched rank-K Gaussian of Lemma 2: every one-dimensional shadow is exactly Gaussian with variance $1 \pm \sqrt{2/K}$. Indeed every single coordinate of the flattened vector is exactly $\mathcal{N}(0,1)$—so a coordinatewise test, one test per scalar logit, would pass the cheat exactly—and the only pathology, the perfect intra-block correlation, is averaged away by one-dimensional projections. The penalty is a fee of order $1/K$. Seen block by block, however, the same configuration is a (translated) line in $\mathbb{R}^C$: support dimension one. Its shadows now have χ^2_1-dispersed scales across directions, and no scalar law can match the fixed target $\mathcal{N}(0,1)$ at two different scales at once. Two devices make this a theorem, uniformly over everything the encoder controls: the modulus of the characteristic function eliminates the constant offset (the arbitrary $\bar\eta$ of the zero-information manifold), reducing the general case to a pure scale family, and the atoms of ρ pin the target at two separated scales. No moment assumptions on the law are used anywhere, so the bound is uniform: a wall, not a fee.

Theorem 3 (Fee versus wall)**.** *Let Assumptions 1–3 hold and let $\mathcal{C}_0$ be the zero-information manifold of Lemma 1. Both statements concern the population functionals equation 20, i.e., the limit of infinitely many batch elements and projections.*

(a) (Fee.) *The law $P^* := \mathcal{N}\big(0, \frac{d}{K}\Pi_G\big)$ lies in $\mathcal{C}_0$—all posteriors uniform, $I(O; Z_k) = 0$ for every k, dynamics loss 0, action sensitivity 0—and*

$$S^{\mathrm{flat}}_\rho(P^*) \;\le\; \bar\kappa_\rho\, \frac{2(d-K)}{K(d+2)} \;\le\; \frac{2\bar\kappa_\rho}{K}, \qquad \textit{hence} \qquad \inf_{P\in\mathcal{C}_0} S^{\mathrm{flat}}_\rho(P) \;\le\; \frac{2\bar\kappa_\rho}{K}. \tag{23}$$

(b) (Wall.) *There is an explicit constant $c_0 = c_0(\rho, C) > 0$, given in equation 25 below—independent of K and of everything the encoder controls—such that*

$$S_\rho^{\mathrm{pd}}(P) \;\geq\; c_0 \qquad \textit{for every } P \in \mathcal{C}_0 \textit{ (arbitrary joint law of } (\mu_1, \dots, \mu_K)\textit{, arbitrary constant } \bar{\eta}). \tag{24}$$

In particular, the exclusion margin of the flattened regularizer vanishes as K grows, while that of the per-distribution regularizer is independent of K.

Proof of (a). Under P^*, $\eta \equiv 0$, so $P^* \in \mathcal{C}_0$ with the listed properties by Lemma 1(ii). The bound is Lemma 2 with $(n, r) = (d, K)$ and $\Pi = \Pi_G$; the projected law is exactly Gaussian because P^* is Gaussian, so equation 22 applies with no remainder. □

Proof of (b). Fix a block k and write $\xi := \mu_k$ (arbitrary law) and $\bar{\eta}^{(k)}$ for the constant centered component. For a fixed direction $v \in \mathbb{S}^{C-1}$, the projected variable is the affine image

$$\langle v, \ell^{(k)} \rangle \;=\; \zeta(v)\, \xi + \beta(v), \qquad \zeta(v) := \tfrac{1}{\sqrt{C}} \langle v, \mathbf{1}_C \rangle, \qquad \beta(v) := \langle v, \bar{\eta}^{(k)} \rangle,$$

with ζ, β deterministic given v. Since $\varphi_{a\xi+b}(t) = e^{ibt} \varphi_\xi(at)$ and the target characteristic function $e^{-t^2/2}$ is real and positive, the reverse triangle inequality $\big|\, |\varphi_X(t)| - e^{-t^2/2} \big| \leq \big| \varphi_X(t) - e^{-t^2/2} \big|$ yields, with $\psi := |\varphi_\xi|$ (even, $\psi \leq 1$),

$$T_\rho\big(\langle v, \ell^{(k)} \rangle\big) \;\geq\; m\big(\zeta(v)\big), \qquad m(a) := \int_{\mathbb{R}} \big(\psi(at) - e^{-t^2/2}\big)^2 \, \mathrm{d}\rho(t) :$$

the constant offset β has dropped out entirely, and m is even and continuous (dominated convergence; the integrand is bounded by 4). Under $v \sim \mathrm{Unif}(\mathbb{S}^{C-1})$, $\zeta^2 \sim \mathrm{Beta}\big(\frac{1}{2}, \frac{C-1}{2}\big)$, so $|\zeta|$ has a density $f_{|\zeta|}$ proportional to $(1 - a^2)^{(C-3)/2}$ on $(0, 1)$: continuous and bounded below on compact subsets of $(0, 1)$.

Suppose $\mathbb{E}_\zeta\, m(\zeta) < \delta$. Set $r := t_1/t_2 \in (0, 1)$, $A := [\frac{1}{4}, \frac{1}{2}]$,

$$f_* := \min_{[r/4,\, 1/2]} f_{|\zeta|} > 0, \qquad \varepsilon := \frac{f_*}{8\,(1 + r^{-1})}.$$

By Markov's inequality, the set $S := \{a > 0 : m(a) \leq \delta/\varepsilon\}$ satisfies $\mathbb{P}(|\zeta| \notin S) < \varepsilon$. Since $f_{|\zeta|} \geq f_*$ on $[r/4, 1/2] \supseteq A \cup rA$, the exceptional sets have small Lebesgue measure: $\mathrm{Leb}(A \setminus S) < \varepsilon/f_*$ and $\mathrm{Leb}(\{a \in A : ra \notin S\}) = r^{-1}\, \mathrm{Leb}(rA \setminus S) < r^{-1}\varepsilon/f_*$. Hence

$$\mathrm{Leb}\big(\{a \in A : a \in S \text{ and } ra \in S\}\big) \;\geq\; \tfrac{1}{4} - (1 + r^{-1})\, \varepsilon/f_* \;=\; \tfrac{1}{8} \;>\; 0,$$

so such an a exists. The atom of mass ρ_1 at t_1 gives, from $m(a) \leq \delta/\varepsilon$, that $|\psi(at_1) - e^{-t_1^2/2}| \leq \sqrt{\delta/(\varepsilon\rho_1)}$; the atom of mass ρ_2 at t_2 gives, from $m(ra) \leq \delta/\varepsilon$ and $(ra)\, t_2 = a\, t_1$, that $|\psi(at_1) - e^{-t_2^2/2}| \leq \sqrt{\delta/(\varepsilon\rho_2)}$. The triangle inequality then forces

$$e^{-t_1^2/2} - e^{-t_2^2/2} \;\leq\; \sqrt{\delta/\varepsilon}\, \big(\rho_1^{-1/2} + \rho_2^{-1/2}\big),$$

i.e., $\delta \geq c_0$ with

$$c_0 \;:=\; \varepsilon\, \frac{\big(e^{-t_1^2/2} - e^{-t_2^2/2}\big)^2}{\big(\rho_1^{-1/2} + \rho_2^{-1/2}\big)^2}. \tag{25}$$

Thus $\mathbb{E}_\zeta m(\zeta) < \delta$ implies $\delta \geq c_0$; hence $\mathbb{E}_v\, T_\rho(\langle v, \ell^{(k)} \rangle) \geq \mathbb{E}_\zeta m(\zeta) \geq c_0$ for every block, and averaging over k gives equation 24. No property of the law of ξ beyond $|\varphi_\xi| \leq 1$ was used, and ε depends only on C and t_1/t_2, so $c_0 = c_0(\rho, C)$ is uniform over $\mathcal{C}_0$. □

Corollary 4 (Landscape reading)**.** *Write the regularized population objective, restricted to the representation-shaping terms as in Remark 1, as $L_{\mathrm{dyn}} + \lambda S$. (i) Under the flattened variant, total collapse—zero mutual information, zero dynamics loss, zero action sensitivity—is available at extra cost at most $2\lambda\bar{\kappa}_\rho/K$ over the unregularized task optimum; consequently, every configuration whose dynamics loss alone exceeds $2\lambda\bar{\kappa}_\rho/K$ is strictly beaten by total collapse. (ii) Under the per-distribution variant, every $P \in \mathcal{C}_0$ incurs total objective at least λc_0; hence any configuration achieving $L_{\mathrm{dyn}} + \lambda S_\rho^{\mathrm{pd}} < \lambda c_0$ is strictly preferred to every totally collapsed one.*

Proof. Immediate from Theorem 3 and Lemma 1(ii). □

Remark 2 (Realizability and the empirical signature). Theorem 3 is stated at the level of logit laws, which is all the regularizer sees. If the observation distribution is atomless and the encoder class is expressive, laws arbitrarily close to P^* are realizable by deterministic encoders. A constant encoder, by contrast, does not realize the cheat: a point mass has trace zero and is heavily penalized under the fixed target (Assumption 2). Matching the trace therefore forces the gauge coordinates to be input-dependent—real input variance laundered through the blind spot—which is exactly the signature of Figure 3: code perplexity at its maximum C (logits vary a lot) with mutual information at zero (probabilities carry nothing).

Remark 3 (Continuous weights). If, instead of atoms, ρ has an absolutely continuous part whose density ϱ is bounded below on $[-t_0, t_0]$ for some $t_0 > 0$ (e.g., the Gaussian weight), the wall holds with the knots replaced by scale bands: choose disjoint compacts $J_1, J_2 \subset (0, 1)$ with $\inf J_2 / \sup J_1 \geq r_0 > 1$; from $\mathbb{E}_\zeta m(\zeta) < \delta$ extract representative scales $g_i \in J_i$ with $m(g_i) \leq \delta / \mathbb{P}(|\zeta| \in J_i)$; the substitution $s = rt$ with $r := g_2/g_1$ (density-ratio constant $\Lambda := \sup \varrho(s/r)/(r\varrho(s)) < \infty$ over the relevant range) transports the g_2-estimate to the g_1 scale, and the triangle inequality in $L^2(\rho|_{[-t_0,t_0]})$ yields $\|e^{-t^2/2} - e^{-r^2t^2/2}\|_{L^2} \leq \sqrt{\delta/p_2} + \sqrt{\Lambda\delta/p_1}$, contradicting $r \geq r_0$ for δ below an explicit threshold. The atomic proof above is this argument with the bands collapsed to knots, and matches the implementation.

Remark 4 (Near-collapse: the wall degrades continuously). If $\mathbb{E}\|\eta^{(k)} - \bar{\eta}^{(k)}\|^2 \leq \epsilon$ for some constant $\bar{\eta}^{(k)}$, then coupling with the collapsed block, $|\varphi_X(t) - \varphi_Y(t)| \leq |t|\sqrt{\mathbb{E}(X-Y)^2}$ and $\mathbb{E}_v\langle v, \eta^{(k)} - \bar{\eta}^{(k)}\rangle^2 = \mathbb{E}\|\eta^{(k)} - \bar{\eta}^{(k)}\|^2/C$ give, by the triangle inequality in L^2 over directions and ρ,

$$\Big(\mathbb{E}_v\, T_\rho\big(\langle v, \ell^{(k)}\rangle\big)\Big)^{1/2} \;\geq\; \sqrt{c_0} \;-\; \Big(\int t^2 \,\mathrm{d}\rho\Big)^{1/2} \sqrt{\epsilon/C}.$$

Escaping the wall requires injecting genuine variance into the centered logits—non-degenerate, input-dependent posteriors—which is precisely the prevention of collapse.

Remark 5 (Why rescaling λ is no substitute for alignment). Since the fee is linear in λ, the rescaling $\lambda \mapsto K\lambda$ makes the fee $O(1)$; it does not close the family of cheats. Lemma 2 equally prices partial collapse: laws with m dead blocks supported on rank $d - m(C-1)$ exist (e.g., the trace-matched Gaussian on the span of the m gauge lines and the remaining $K - m$ full blocks; its m collapsed blocks carry uniform posteriors and zero information) with flattened penalty at most $2\bar{\kappa}_\rho\, m(C-1)/\big((d - m(C-1))(d+2)\big)$. The first dead block thus costs the flattened test only about $2\bar{\kappa}_\rho(C-1)/d^2$ ($\approx 6 \times 10^{-5}$ at our configuration), a further factor $\sim C$ below the per-block share of the total-collapse fee, so the rescaling required to price a cheat grows with its granularity and no single constant covers the family. The per-distribution wall, whose proof is per block, instead charges every dead block $\lambda c_0/K$ uniformly, regardless of the state of the others. Moreover, the empirical statistic has a null floor $\Theta(1/B_1)$ (Remark 8) which rescaling amplifies together with the signal, while the flattened discriminative gap already sits below that floor at our batch size. Empirically, both variants exhibit a sharp critical weight on Kung-Fu Master—$\lambda^*_{\mathrm{pd}} \in (0.05, 0.08]$ versus $\lambda^*_{\mathrm{flat}} \in (1.0, 1.2]$ (Figure 5), a ratio within $[12.5, 24]$, consistent with the ratio $\approx 0.063/0.0055 \approx 11.5$ between the per-distribution and flattened prices of the realized cheat (Remark 8). The deficiency of the flattened test is one of alignment, not of magnitude—the content of Proposition 5.

J.5 Alignment is necessary: blocking has no intrinsic power

Intuition. If per-distribution slicing were simply a stronger test, one could hope to fix the flattened variant by blocking arbitrarily. It is not: rotate the very same rank-K structure out of alignment with the blocks and each block sees a trace-matched rank-K marginal, so by Lemma 2 the per-block test is exactly as blind as the flattened one. The power of the per-distribution formulation comes entirely from the softmax pinning the task-free subspace to one known direction per block: the test is sliced exactly where the architecture's invariance lives.

Proposition 5 (Rotated degeneracies defeat block slicing)**.** *Assume $K \leq C$ and let $H \in O(K)$ be an orthogonal matrix with $|H_{jk}| = K^{-1/2}$ for all j, k (e.g., a normalized Hadamard matrix; $K = 16$ in our configuration). Let $G' \subset \mathbb{R}^d$ be spanned by the orthonormal vectors $f_j := \sum_{k=1}^K H_{jk}\, e_{k,j}$, where $e_{k,j}$ denotes the j-th coordinate vector of block k, and let $P' := \mathcal{N}\big(0, \frac{d}{K}\Pi_{G'}\big)$ be the rotated*

counterpart of the law P^* of Theorem 3(a). Then every block marginal of P' is $\mathcal{N}\big(0, \frac{C}{K}\Pi_K\big)$ with Π_K a rank-K projector on $\mathbb{R}^C$, and consequently

$$S_\rho^{\mathrm{pd}}(P') \;\le\; \bar{\kappa}_\rho \,\frac{2(C-K)}{K(C+2)} \;\le\; \frac{2\bar{\kappa}_\rho}{K}. \tag{26}$$

Proof. Write $X = \sum_j \alpha_j f_j$ with $\alpha \sim \mathcal{N}\big(0, \frac{d}{K} I_K\big)$. The block-$k$ component of f_j is $H_{jk}\, e_j^{(k)}$ (the j-th coordinate vector of $\mathbb{R}^C$), so $X^{(k)} = \sum_{j \le K} \alpha_j H_{jk}\, e_j^{(k)}$ has independent Gaussian coordinates with variance $\frac{d}{K} H_{jk}^2 = \frac{d}{K^2} = \frac{C}{K}$ on its first K coordinates and 0 elsewhere: a trace-matched ($\mathrm{tr} = C$) isotropic rank-K Gaussian on $\mathbb{R}^C$. Apply Lemma 2 with $(n, r) = (C, K)$ to each block and average. □

Remark 6 (Reading of Proposition 5). P' is not supported on the gauge subspace, hence not task-free: this is a statement about the test's power, not about a cheat available to the encoder under per-distribution regularization. Haar-generic rotations behave like the explicit H (principal cosines concentrate; Monte Carlo: per-block excess 0.0645 aligned versus 0.0051 rotated, matching the flattened value). Together with Theorem 3 this fixes the design principle—sketched-isotropy regularization must be applied at the level at which the architecture defines its invariances: here, one test per categorical distribution—and delimits what transfers to architectures without a distinguished subspace: against a generically placed rank-r degeneracy, fixed blocks buy nothing, and the $1/r$ certificate of Lemma 2 is all a sketched test provides.

J.6 Complements

Remark 7 (Mutual information at the target). Fix $C \ge 2$ and define $m_C^* := \big(\log C - \mathbb{E}[H(\mathrm{softmax}(G))]\big)/\log C$ with $G \sim \mathcal{N}(0, I_C)$: the normalized mutual information of equation 9 evaluated at the target law ($m_{96}^* \approx 0.106$ by Monte Carlo). The flattened variant certifies no positive mutual information—Theorem 3(a) exhibits zero-information laws with vanishing penalty—and the perplexity of equation 10 cannot separate the target from uniform collapse (it evaluates to C in both cases), which is why Figure 3 pairs it with mutual information.

Remark 8 (Numerical anchors and finite-sample effects). Fee: the crude global constant gives $2\bar{\kappa}_\rho/K = 0.094$ at $K = 16$ (Gaussian weight); a second-order expansion of $T_\rho(\mathcal{N}(0, V))$ around $V = 1$ replaces $\bar{\kappa}_\rho$ by $\kappa_\rho = \frac{1}{12\sqrt{3}}$, predicting $2\kappa_\rho/K = 0.0060$, in agreement with the Monte Carlo value 0.0055 at $(K, C, B_1) = (16, 96, 32)$; the measured excess over the finite-batch null floor follows $\approx 2\kappa_\rho/r$ across $r \in \{1, \ldots, 256\}$ and is invariant in the ambient dimension at fixed r (e.g. $\mathrm{sd}(V) = 0.358$ measured versus 0.352 exact at $(n, r) = (1536, 16)$); the second-order rule is accurate for $r4$ only (at $r = 1$ the exact value $\int T_\rho(\mathcal{N}(0, V))\, \chi_1^2(\mathrm{d}V) = 0.067$ differs from $2\kappa_\rho = 0.096$). Wall: the constant c_0 of equation 25 is a qualitative guarantee and is far from sharp (the proof samples $|\zeta|$ on $[\frac{1}{4}, \frac{1}{2}]$, in the tail of its law; at $C = 96$ it is of order 10^{-9} for our weight). The collapsed law realized in training—Gaussian gauge coordinates with $\mathrm{Var}(\mu_k) = C$, every logit coordinate exactly $\mathcal{N}(0, 1)$ (Remark 2)—pays a per-distribution penalty of $\approx 6 \times 10^{-2}$ (χ_1^2-dispersed shadow variances), roughly $11\times$ the flattened fee it pays: the order-K separation the two bounds predict. Other zero-information laws can pay less than this cheat, but never less than c_0. At the paper's configuration $(\lambda, K, C, B_1) = (0.1, 16, 96, 32)$: fee $\le \lambda \cdot 0.094$ rigorously, $\approx \lambda \cdot 0.0055 = 5.5 \times 10^{-4}$ as measured; per-distribution price of the realized cheat $\approx \lambda \cdot 0.063 = 6.3 \times 10^{-3}$ as measured. A different (e.g., knotted) normalization rescales both numbers identically, leaving the order-K ratio—the invariant content of Theorem 3—unchanged. Finite B_1 and M: all kernels in the closed form of the Epps–Pulley statistic are bounded, so Hoeffding-type bounds for bounded U/V-statistics give deviations $O(M^{-1/2} + B_1^{-1/2})$ around the population values, with an empirical null floor $\Theta(1/B_1)$ (≈ 0.013 at $B_1 = 32$, matching Monte Carlo); neither quantity interacts with the $1/K$ fee, which is a population bias, not noise (Figure 14: sample starvation was never the mechanism).

Remark 9 (Scope). These are statements about the loss landscape, not about the optimization path; that SGD indeed finds the spurious optimum under flattened regularization, and avoids it under per-distribution regularization, is established empirically (Figures 3, 4, 13 and 14). Both variants converge to comparable values of their own statistic (Figure 3), which is exactly what the theory predicts: the statistic's value does not discriminate; what it can see does. The theorem also concerns only the marginal law of the latents: it explains why per-distribution SIGReg prevents static collapse without

a stop-gradient, and is silent on the conditional (action-dependent) structure that the stop-gradient preserves (Appendix B)—consistent with the concluding remark of the main text that removing the stop-gradient entirely will require regularizing the conditional law. Beyond this architecture, in settings without a task-free subspace (continuous JEPA embeddings), rank-r degeneracy is neither free nor information-less, so the $1/r$ soft spot can manifest at most as a soft equilibrium at reduced effective rank; it is the gauge structure of the softmax that converts the statistical soft spot into a zero-cost tunnel, and the per-distribution formulation is the aligned test that walls it off.