Weight a moment: risk-aware exploration with Bayes

Karim Zaghw, Peter Dayan, Georgy Antonov

Keywords: Bayesian Q-learning, distributional reinforcement learning, risk, exploration, value of perfect information (VPI), moment/mean embeddings

Summary

Distributional reinforcement learning (DRL) conventionally models objective uncertainty in the return distributions from states, allowing important quantities to be computed such as risk-sensitive policies. Bayesian approaches additionally quantify subjective, or epistemic, uncertainty and can therefore provide principled signals for exploration. However, most Bayesian value-based reinforcement learning algorithms, such as the seminal Bayesian Q-learning (BQL), rely on restrictive likelihood assumptions to remain tractable. This severely constrains the set of approximating return distributions, and thus the risk-relevant statistics they can represent. Due to these limitations, Bayesian risk-sensitive exploration has been relatively underexplored. We propose a Bayesian DRL algorithm which maintains and updates a full posterior belief over finite-dimensional mean embeddings of the return distribution, chosen to capture higher-order moments and admit Bellman-consistent propagation. This representation allows us to generalize the value of perfect information exploration policy that BQL employs to moment-based risk objectives, enabling exploration that explicitly trades off epistemic uncertainty and risk. As we show in simulations, our algorithm learns more accurate approximations to return distributions compared to BQL, and the resulting risk-sensitive exploration improves control performance over a matched non-Bayesian DRL baseline.

Contribution(s)

1. We introduce Bayesian Moment Learning (BML), a model-free DRL algorithm which maintains a Bayesian posterior over finite-dimensional mean embeddings of return distributions. We show that BML can approximate (and model epistemic uncertainty over) a richer class of return distributions compared to BQL.
   Context: BML builds on mean-embedding sketches and kernel mean-embedding (Wenliang et al., 2023; Rowland et al., 2019; Muandet et al., 2017), and thereby relaxes the Gaussian-return restriction that underpins BQL (Dearden et al., 1998). The derived Bayesian updates performed by BML extend sketch-space temporal difference learning with BQL-inspired mixture updating and projection to preserve a tractable approximate posterior family.
2. BML explicitly represents epistemic uncertainty over important risk-relevant statistics (distribution moments). We use this uncertainty to generalize the value-of-perfect-information (VPI) exploration policy to risk-aware objectives. We show in simulations using a widely-adopted risky control task that BML exhibits efficient and risk-sensitive exploration.
   Context: This action-selection rule is rooted in the classical VPI principle from Bayesian RL and decision analysis (Dearden et al., 1998; Howard, 1966), but targets risk-sensitive criteria rather than expected return alone. We focus on objectives that are computable/approximable from finitely many moments.

Weight a moment: risk-aware exploration with Bayes

Karim Zaghw[1,2], **Peter Dayan**[1,2], **Georgy Antonov**[1,2]
karim.zaghw@student.uni-tuebingen.de, dayan@tue.mpg.de,
ga2578@princeton.edu

[1]**Max Planck Institute for Biological Cybernetics, Tübingen, Germany**
[2]**University of Tübingen, Germany**

Abstract

Distributional reinforcement learning (DRL) conventionally models objective uncertainty in the return distributions from states, allowing important quantities to be computed such as risk-sensitive policies. Bayesian approaches additionally quantify subjective, or epistemic, uncertainty and can therefore provide principled signals for exploration. However, most Bayesian value-based reinforcement learning algorithms, such as the seminal Bayesian Q-learning (BQL), rely on restrictive likelihood assumptions to remain tractable. This severely constrains the set of approximating return distributions, and thus the risk-relevant statistics they can represent. Due to these limitations, Bayesian risk-sensitive exploration has been relatively underexplored. We propose a Bayesian DRL algorithm which maintains and updates a full posterior belief over finite-dimensional mean embeddings of the return distribution, chosen to capture higher-order moments and admit Bellman-consistent propagation. This representation allows us to generalize the value of perfect information exploration policy that BQL employs to moment-based risk objectives, enabling exploration that explicitly trades off epistemic uncertainty and risk. As we show in simulations, our algorithm learns more accurate approximations to return distributions compared to BQL, and the resulting risk-sensitive exploration improves control performance over a matched non-Bayesian DRL baseline.

1 Introduction

Uncertainty in reinforcement learning (RL) (Sutton et al., 1998) comes in at least two forms. Aleatoric uncertainty reflects the inherent randomness of returns induced by stochastic dynamics and rewards, as well as the decision policy. Epistemic uncertainty reflects what the agent does not yet know due to limited priors and data. Both matter: epistemic uncertainty determines potentially costly or risky exploration, while aleatoric uncertainty determines additional, irreducible, risk. However, classic, value-based RL typically ignores both, collapsing the future into a single number, the expected long-run, discounted utility, i.e., the expected return.

The most complete accounting of the uncertainty that matters for reinforcement is the entire distribution of returns. From the perspective of aleatoric uncertainty, this provides access to variability, tails, and higher-order shape information, and can naturally support computations such as risk-sensitive decision-making (Heger, 1994). In many applications such as robotics, operations research, finance, and safety-critical control, the difference between good on average and bad in the worst case is decisive. Moreover, risk preferences vary across tasks and users, motivating methods that can trade off performance against downside risk.

From the perspective of epistemic uncertainty, the focus has been on exploration. Exploration is an information-gathering process: the agent should seek actions that reduce uncertainty where doing so is valuable for future decisions. Bayesian RL offers a principled framework for doing this by

maintaining beliefs over value-related quantities, updating those beliefs with experience, and selecting actions based on the expected benefit of information. Bayesian Q-Learning (BQL) (Dearden et al., 1998) is a canonical model-free instance of this idea: it maintains approximate posterior distributions over action values and uses a value-of-perfect-information (VPI) criterion to quantify an approximation to the expected improvement in the return from exploration (Howard, 1966).

However, existing RL approaches sit awkwardly across the aleatoric/epistemic divide. On the Bayesian side, tractability often comes from restrictive likelihood assumptions, constraining the class of return distributions and thus the risk-relevant statistics that can be represented. Conversely, distributional reinforcement learning (DRL) (Bellemare et al., 2017) has concentrated on aleatoric uncertainty, and typically does not maintain a Bayesian posterior that captures the epistemic uncertainty necessary for principled exploration. Moreover, even when information-value criteria such as VPI are used, they are commonly focused on improving the expected return rather than general risk-sensitive objectives, leaving Bayesian risk-sensitive exploration comparatively underdeveloped.

This paper aims to close that gap. We propose a Bayesian distributional RL approach called Bayesian Moment Learning (BML) that maintains a posterior over a finite-dimensional mean embedding of the return distribution. By choosing embeddings that capture higher-order moments and admit Bellman-consistent propagation (Wenliang et al., 2023), we obtain a computationally practical belief state that represents risk-relevant return structure, together with epistemic uncertainty about it. We use this belief state to generalize VPI-style action selection from expected return to moment-based risk objectives, enabling exploration that explicitly trades off epistemic uncertainty against risk. Empirically, BML shows improved approximation of return distributions in policy evaluation and efficient risk-sensitive exploration that improves control under a risk-aware objective.

2 Background

We consider an agent interacting with an environment modeled as a Markov decision process (MDP) (Puterman, 2014) $\langle \mathcal{S}, \mathcal{A}, R, T, \gamma \rangle$, with state space $\mathcal{S}$, action space $\mathcal{A}$, discount factor $\gamma \in [0, 1)$ (with $\gamma = 1$ used only in finite-horizon episodic settings such as in Section 4.2.1), transition kernel to next states $T : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathcal{S})$, and stochastic reward kernel over real-valued rewards $R : \mathcal{S} \times \mathcal{A} \to \mathcal{P}(\mathbb{R})$, where $\mathcal{P}(\cdot)$ denotes the set of probability measures over the given space. A policy π maps each state to a distribution over actions, $\pi : \mathcal{S} \to \mathcal{P}(\mathcal{A})$.

The discounted random return from $\{S_0, A_0\} = \{s, a\}$ under policy π is $Z^\pi(s, a) := \sum_{t=0}^{\infty} \gamma^t R_t \mid \{S_0 = s, A_0 = a\}$. The corresponding return distribution is the law of this random variable: $\eta^\pi(s, a) := \text{Dist}(Z^\pi(s, a)) \in \mathcal{P}(\mathbb{R})$. The conventional state-action value function is the expected return, $Q^\pi(s, a) := \mathbb{E}[Z^\pi(s, a)]$, and the associated state value function is $V^\pi(s) := \mathbb{E}^\pi[Q^\pi(s, A)]$. The Bellman operator for policy π acting on any Q is defined as $(\mathcal{T}^\pi Q)(s, a) := \mathbb{E}^\pi \left[R + \gamma \mathbb{E}^\pi \left[Q(S', A')\right]\right]$, so that Q^π satisfies the Bellman equation $Q^\pi = \mathcal{T}^\pi Q^\pi$. A *Bellman update* applies (an approximation of) this operator for each realized transition $\langle s_t, a_t, r_t, s_{t+1} \rangle$, e.g. the sample-based Q-learning update $Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \alpha_t \left(r_t + \gamma \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)\right)$ where α_t is a learning rate (Watkins & Dayan, 1992).

In the standard risk-neutral control objective, we seek a policy, π^*, that maximizes expected return: $\pi^\star \in \arg\max_\pi \; V^\pi(s)$ for all $s \in \mathcal{S}$. In this paper, we also consider objectives that depend on the *shape* of $\eta^\pi(s, a)$, rather than only its mean, as we make precise in later sections.

2.1 Distributional Reinforcement Learning

Distributional reinforcement learning (DRL) famously treats the random return as the primary object of estimation and calculation and aims to learn its *full distribution* (Dabney et al., 2018; Bellemare et al., 2017). A central characterization of η^π is the distributional Bellman equation:

$$Z^\pi(s, a) \overset{D}{=} R(s, a, S') + \gamma Z^\pi(S', A') \;, \text{ with } S' \sim T(\cdot \mid s, a), A' \sim \pi(\cdot \mid S') \tag{1}$$

where $\overset{D}{=}$ denotes equality in distribution. This induces the corresponding distributional Bellman operator on return, which is the target of DRL's canonical suite of temporal difference learning rules. The distributional view captures aleatoric uncertainty in the form of the intrinsic randomness of returns induced by the underlying MDP, and is conceptually distinct from *epistemic* uncertainty about potentially knowable quantities (Bellemare et al., 2017).

Implementing DRL typically requires a finite-dimensional representation of $\eta^\pi(s,a)$ and a learning rule that approximates the distributional Bellman update. One prominent approach is the *categorical* representation (e.g., C51), which approximates $\eta^\pi(s,a)$ by a discrete distribution supported on a fixed set of atoms and learns their probabilities using a projected Bellman target and a divergence-based loss (Bellemare et al., 2017). Another approach is the *quantile* representation, where the distribution is approximated by a uniform mixture of learnable quantile locations (Dabney et al., 2018).

2.2 Mean-Embedding Sketches and the Sketch Bellman Operator

Following work in probabilistic population codes (Sahani & Dayan, 2003), we consider a different finite representation of $\eta^\pi(s,a)$ involving mean-embedding *sketches* (Rowland et al., 2019; Bellemare et al., 2023), which are closely related to kernel mean embeddings in traditional machine learning (Muandet et al., 2017). These represent a distribution $\eta \in \mathcal{P}(\mathbb{R})$ through the expectation of a multidimensional feature map $\boldsymbol{\phi} : \mathbb{R} \to \mathbb{R}^m$, i.e. $\mathbb{E}_{Z\sim\eta}[\boldsymbol{\phi}(Z)]$.

The (mean-embedding) sketch of the return distribution for a state-action pair is defined as $\mathbf{U}^\pi(s,a) := \mathbb{E}[\boldsymbol{\phi}(Z^\pi(s,a))] = \mathbb{E}_{Z\sim\eta^\pi(s,a)}[\boldsymbol{\phi}(Z)]$. Wenliang et al. (2023) showed that the Bellman update can be carried out directly in embedding space for suitably chosen feature maps; this avoids the need for the expensive "imputation" step (decoding sketches back into distributions). That is, if for each reward value r there is a matrix $B_r \in \mathbb{R}^{m\times m}$ such that $\boldsymbol{\phi}(r+\gamma g) = B_r\,\boldsymbol{\phi}(g)$ for all g, then the sketch satisfies a closed Bellman-style recursion: $\mathbf{U}^\pi(s,a) = \mathbb{E}[B_R\,\mathbf{U}^\pi(S',A')]$. This motivates the *sketch Bellman operator* $(T^\pi_\phi \mathbf{U})(s,a) := \mathbb{E}[B_R\,\mathbf{U}(S',A')]$. Notably, Wenliang et al. (2023) observe that any feature map spanning degree-k polynomials is Bellman-closed. In Section 3.1 we use polynomial sketches but drop the constant feature (its embedding coordinate is deterministically 1), yielding the equivalent affine form $\boldsymbol{\phi}(r+\gamma g) = B_r\,\boldsymbol{\phi}(g) + \boldsymbol{\phi}(r)$ (see Supplementary Section 6).

We build on this framework by (i) choosing a feature map which leads to embeddings which enjoy Bellman-closedness and capture information about higher-order moments of the return distributions, and (ii) maintaining a full *Bayesian posterior* over $\mathbf{U}(s,a)$ (dropping π where this is clear) which, as we show later, is naturally amenable to risk-aware exploration heuristics.

2.3 Bayesian Q-Learning (BQL)

Bayesian Q-learning (BQL) is a notable temporal difference (TD) DRL algorithm which explicitly represents epistemic uncertainty about action values (Dearden et al., 1998). BQL assumes that the return for each (s,a) pair is Gaussian with unknown mean and precision: $Z(s,a) \sim \mathcal{N}(\mu_{s,a}, \tau^{-1}_{s,a})$, where $\mu_{s,a} = Q(s,a)$. The prior over $(\mu_{s,a}, \tau_{s,a})$ is further assumed to belong to the conjugate Normal-Gamma family. Epistemic uncertainty about $Q(s,a)$ is therefore encoded as $p(\mu_{s,a})$.

The TD learning rule constructs a bootstrap target from each transition (s,a,r,s'), i.e., $r+\gamma Z(s',a')$ (often with greedy a' at s'). Because BQL maintains a belief over the full return $Z(s,a)$, bootstrapping requires marginalizing uncertainty in the next-state return, which induces a non-conjugate mixture posterior. To preserve a Normal–Gamma belief representation, BQL projects this mixture back onto the Normal–Gamma family via KL minimization (Dearden et al., 1998). We discuss the details of this projection step in the case of BML in Supplementary Section 7.7.

Given belief $p(\mu_{s,a})$, BQL uses a myopic value-of-perfect-information (VPI) criterion to pick actions. The exploratory policy takes the form $a \in \arg\max_b \{\mathbb{E}[\mu_{s,b}] + \mathrm{VPI}(s,b)\}$, where $\mathrm{VPI}(s,b)$ denotes the expected improvement in decision quality if the true value of action b were revealed (Dearden et al., 1998).

3 Bayesian Moment Learning

At the core of BQL is the assumption that return is Gaussian. This severely limits the scope of the approximating distributions learnt by BQL, and as a result, the veracity of epistemic uncertainty it represents (Antonov & Dayan, 2024). In Bayesian Moment Learning (BML), we improve the representational capacity of BQL by maintaining a belief in the embedding space generated by polynomial features, which correspond to raw, uncentred moments of the return distribution.

3.1 Bellman-Closed Polynomial Embeddings

We choose polynomial features so that the sketch Bellman operator is closed in embedding space (Wenliang et al., 2023), however we drop the constant term such that $\boldsymbol{\phi}(z) = (z, z^2, \dots, z^k)^\top$. Then for each reward r we can construct a matrix $B_r \in \mathbb{R}^{k\times k}$ such that $\boldsymbol{\phi}(r + \gamma z') = B_r\boldsymbol{\phi}(z') + \boldsymbol{\phi}(r)$ for all z' (see Supplementary Section 6). This property yields an embedding-space bootstrap target for our Bayesian update and avoids the need for decoding sketches back into distributions.

3.2 Bayesian Model in Embedding Space

Critically, unlike Wenliang et al. (2023) (but like BQL's belief over the mean Q-value $\mu_{s,a}$), we maintain a Bayesian belief over the mean embedding $\mathbf{U}(s,a) = \mathbb{E}[\boldsymbol{\phi}(Z(s,a))]$. With $\mathbf{Y}(s,a) = \boldsymbol{\phi}(Z(s,a)) \in \mathbb{R}^k$ as the embedded return, we model an observation $\mathbf{y} \in \mathbb{R}^k$ as conditionally Gaussian:

$$p(\mathbf{y} \mid \mathbf{U}(s,a), \Lambda(s,a)) = \mathcal{N}\big(\mathbf{y};\, \mathbf{U}(s,a),\, \Lambda(s,a)^{-1}\big), \tag{2}$$

with a conjugate Normal–Wishart prior on the unknown mean and precision

$$(\mathbf{U}(s,a), \Lambda(s,a)) \sim \mathrm{NW}(\boldsymbol{\mu}_0, \lambda, W, \nu), \tag{3}$$

where $\Lambda(s,a) \sim \mathcal{W}(W,\nu)$, and $\mathbf{U}(s,a) \mid \Lambda(s,a) \sim \mathcal{N}\big(\boldsymbol{\mu}_0, (\lambda\Lambda(s,a))^{-1}\big)$. Marginalizing out $\Lambda(s,a)$ yields a multivariate Student-t belief over $\mathbf{U}(s,a) \in \mathbb{R}^k$ (Supplementary Section 7.3).

This Normal–Wishart model yields closed-form posteriors for $\mathbf{U}(s,a)$ given i.i.d. observations in embedding space, allowing us to implement Bayesian updates efficiently. However, since TD's bootstrapped targets are not i.i.d. samples of the true embedded return, conjugacy only holds for terminal transitions; all other updates require integrating over next-state uncertainty, producing a non-conjugate mixture posterior that we approximate using BQL-style mixture-and-projection steps (Supplementary Section 7.7). Moreover, because $\boldsymbol{\phi} : \mathbb{R} \to \mathbb{R}^k$, valid mean embeddings occupy a constrained subset of $\mathbb{R}^k$ that our Gaussian embedding likelihood does not enforce explicitly (we examine this caveat in Supplementary Section 12). In the main text, we focus instead on convergence of the learned moments to the true return moments (Section 4.1.1), which we view as sufficient evidence that the approximation behaves well in practice. In summary, we treat the Gaussian embedding likelihood as a pragmatic approximation enabling efficient Bayesian inference and exploration in moment space; we validate it via moment convergence in the main text and report feasibility diagnostics in Supplementary Section 12.

Note that on terminal transitions (i.e., when the update is a direct observation in embedding space rather than a bootstrapped target), the update to the *first-moment* component performed by BML coincides exactly with BQL's update to Q. This agreement is visible in the moment-convergence results in Figure 1 and demonstrated formally in Supplementary Section 9.1. The mixture posterior and KL projection breaks this exact equivalence for all other states, leading to the observed divergence in first-moment predictions (Figure 1).

3.2.1 Mixture Posterior Update in Embedding Space

Let a' denote the next action used for bootstrapping (for example, greedy with respect to the current objective), and define the random next embedded return $\mathbf{Y}' := \boldsymbol{\phi}(Z(s',a'))$. The distribution of $\mathbf{Y}'$ is induced by our current belief at (s',a').

According to the distributional Bellman equation (1), the embedding-space target for TD updates is $\boldsymbol{\phi}(r+\gamma Z(s',a'))$. For Bellman-closed embeddings we can compute this target directly from $\mathbf{Y}'$. For polynomial features (Section 3.1) we have the exact relationship $\boldsymbol{\phi}(r+\gamma Z(s',a')) = B_r\,\mathbf{Y}' + \boldsymbol{\phi}(r)$.

Conditioned on a particular realization $\mathbf{y}$ of $\mathbf{Y}'$, the bootstrap target $\mathbf{x} = B_r\mathbf{y} + \boldsymbol{\phi}(r)$ acts as a pseudo-observation for $\boldsymbol{\phi}(Z(s,a))$ under our Gaussian embedding likelihood. The corresponding conditional posterior is conjugate: $p(\mathbf{U}(s,a), \Lambda(s,a) \mid \mathbf{x}) \in \mathrm{NW}$. However, $\mathbf{Y}'$ is random under our current belief, so the correct update integrates over $\mathbf{Y}'$, yielding a mixture posterior

$$p_{\mathrm{mix}}(\mathbf{U}(s,a), \Lambda(s,a) \mid s,a,r,s') \;=\; \int_{\mathbb{R}^k} p(\mathbf{U}(s,a), \Lambda(s,a) \mid \mathbf{x} = B_r\mathbf{y} + \boldsymbol{\phi}(r))\; p_{\mathbf{Y}'}(\mathbf{y})\; d\mathbf{y}, \quad (4)$$

where $p_{\mathbf{Y}'}$ is the predictive distribution of $\mathbf{Y}'$ induced by the belief at (s',a'). Since p_{mix} is generally not Normal–Wishart, we approximate it by a Normal–Wishart belief via a forward-KL projection (sufficient-statistic matching); details are given in Supplementary Section 7.7.

3.3 Risk-Sensitive Action Selection via VPI

Our posterior induces uncertainty over the moment embedding $\mathbf{U}(s,a)$, which we convert into a scalar risk-sensitive score $\Psi(s,a) = \rho\big(\mathbf{U}(s,a)\big)$, where ρ is a moment-based functional, e.g., a mean–dispersion criterion (Mannor & Tsitsiklis, 2011). To avoid undefined quantities when sampled moments are temporarily inconsistent early in training (Supplementary Section 12), we use always-real transforms. With $m_i = \mathbb{E}[Z^i]$ and $v = m_2 - m_1^2$, we define

$$\Psi = m_1 - \lambda\,\sigma(v), \qquad \text{where } \sigma(v) = \mathrm{sign}(v)\sqrt{|v|} \quad (5)$$

Empirically, retaining the sign in $\sigma(v)$ (treating transient $v < 0$ as a moment-feasibility artifact, rather than clipping at $v \geq 0$) avoids an artificially pessimistic early distribution over Ψ that can suppress exploration under high posterior uncertainty. This effect diminishes as learning drives samples toward moment-feasibility (see Supplementary Section 12 for diagnostics). Higher-moment terms can also be included without changing the inference procedure.

rsVPI over Ψ. To distinguish this risk-scored variant from the standard VPI used in BQL (computed over expected return), we refer to it as *risk-sensitive VPI (rsVPI)*. At state s, let $a_1 = \arg\max_a \mathbb{E}[\Psi(s,a)]$ and $a_2 = \arg\max_{a \neq a_1} \mathbb{E}[\Psi(s,a)]$. We define rsVPI for action a as

$$\mathrm{rsVPI}(s,a) = \begin{cases} \mathbb{E}\big[(\mathbb{E}[\Psi(s,a_2)] - \Psi(s,a))_+\big], & a = a_1, \\ \mathbb{E}\big[(\Psi(s,a) - \mathbb{E}[\Psi(s,a_1)])_+\big], & a \neq a_1, \end{cases} \quad (6)$$

where $(x)_+ = \max(x,0)$, and select actions by $a \in \arg\max_b \{\mathbb{E}[\Psi(s,b)] + \mathrm{rsVPI}(s,b)\}$. We use a Monte-Carlo estimator for rsVPI (see Supplementary Section 9.2).

4 Results

We evaluate our Bayesian mean-embedding approach along two axes. First, in *policy evaluation*, we ask whether maintaining a posterior over moment embeddings yields a more accurate approximation of the return distribution than BQL. We assess this via (i) the quality of reconstructed return distributions (Section 4.1.2) and (ii) the tracking and convergence of individual moments (Section 4.1.1).

Second, in *control* (Section 4.2), we test whether Bayesian uncertainty in moment space improves potentially risk-sensitive exploration relative to a non-Bayesian counterpart with the same representational capacity. Concretely, we compare BML against Sketch-TD (Wenliang et al., 2023) with the same polynomial feature map in the BART environment, using the risk score Ψ in (5) and sweeping $\lambda \in \{0,1,2\}$ to cover both risk-neutral and risk-sensitive regimes. To contextualize the risk-neutral case ($\lambda = 0$) against a standard Bayesian baseline, we also include BQL as a reference.

In the control experiments, unless stated otherwise, we instantiate the risk score Ψ using only the first two moments (mean and dispersion) to keep the objective simple and to isolate the effect of Bayesian uncertainty. In Supplementary Section 13.2.1 we additionally extend Ψ with a third-moment term for skewness and show that BML can reliably prefer positively skewed paths over negatively skewed alternatives even when mean and variance are matched.

4.1 Policy Evaluation

To isolate the effect of moment propagation and make distributional differences easy to interpret, we consider a simple chain MDP. The agent starts in state s_0, transitions deterministically to s_1, s_2, s_3, and finally to a terminal state s_4 where it receives a stochastic terminal reward drawn from a mixture of Beta distributions (see Supplementary Section 11.1.2). Thus, all upstream return distributions are nontrivial discounting-based shifts of the same underlying multimodal terminal reward, so any discrepancy in learned moments directly reflects propagation and approximation errors. Furthermore, to control for the effect of prior choice, we initialize BQL and BML with matched priors on the scalar mean (for full details see Supplementary Section 11.1.1). Moreover, within each run both agents are trained on the exact same sampled rewards.

4.1.1 Moment Convergence

We compute the true moments of the return distribution at each state and compare them to the moment estimates produced by BQL and our BML method. For each moment order, we report the absolute relative error (see Supplementary Section 11.1.3). Figure 1 shows that at the terminal state the first-moment updates of BQL and BML coincide (top-right highlighted panel), as mentioned in Section 3.2 and proved in Supplementary Section 9.1. For all other states, the learning signal must be propagated through multiple bootstrapped updates, and, owing largely to the projection step, the BQL and BML estimates deviate, though not severely.

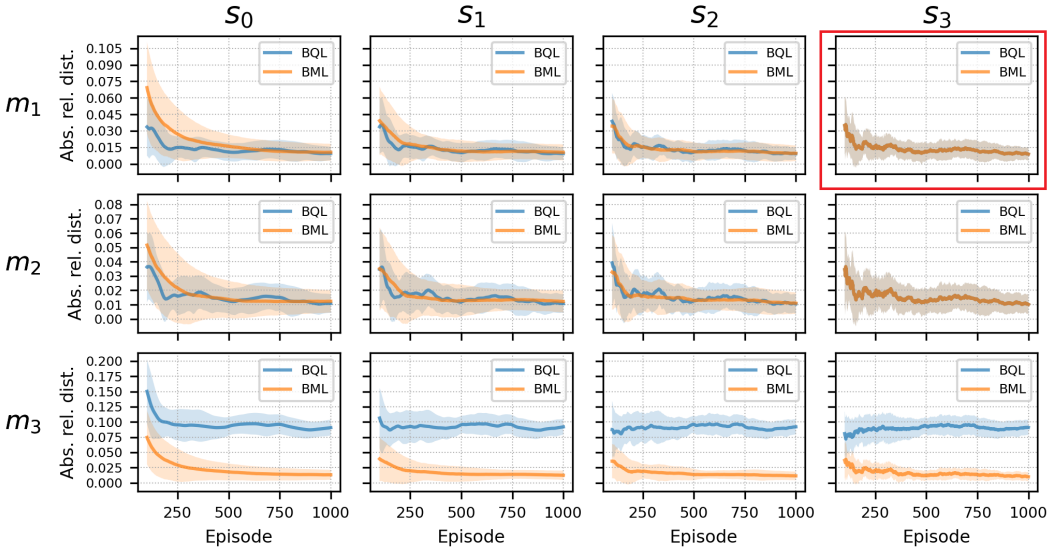


Figure 1: Absolute relative distance between true and estimated return moments for BQL and BML during policy evaluation. Panels from top to bottom correspond to 1st-3rd moments respectively. Top-right highlighted panel shows the terminal-state first-moment updates, which are mathematically identical for BQL and BML. The first 100 episodes are omitted to reduce the visual impact of the initial transient due to prior initialization.

The difference becomes much more pronounced for higher-order moments which BML (but not BQL) models explicitly. In Supplementary Section 13.1.1 we demonstrate the same effect for higher-

order moments beyond the third moment, where the divergence persists and typically grows with moment order, indicating that the embedding-based posterior better preserves higher-order structure of complex return distributions during propagation. We additionally show that this advantage is robust across a range of prior initializations, with BML consistently producing more accurate moment estimates than BQL.

4.1.2 Imputation

To visualize the return distributions implied by BML, we reconstruct (impute) an explicit distribution from the learned representation. Because a finite-dimensional moment embedding does not uniquely identify a return distribution, we treat these imputations as an illustrative decoding of the learned representation; the moment-convergence results (Section 4.1.1) provide the primary, representation-level evidence for the accuracy of BML's approximation. In particular, we use the imputation procedure of Wenliang et al. (2023), which finds a categorical distribution whose mean embedding best matches a target embedding. As the target embedding for BML we use the posterior mean $\mathbb{E}[\mathbf{U}(s,a)]$ with a $k = 10$ polynomial embedding (the first 10 raw moments), yielding a single representative return distribution for comparison. Supplementary Section Supplementary Section 13.1.3 studies the sensitivity of the imputation to the choice of k, and Supplementary Section Supplementary Section 13.1.2 visualizes how posterior-sampled imputations evolve and concentrate over training.

Figure 2 contrasts the distributions learned by BQL and BML. Since BQL is tied to a restrictive parametric form, it does not capture the multimodal structure of the true return distribution. By contrast, the BML embedding can represent higher-order structure through its moments, and the imputed distribution provides a substantially closer qualitative match to the multimodal target.

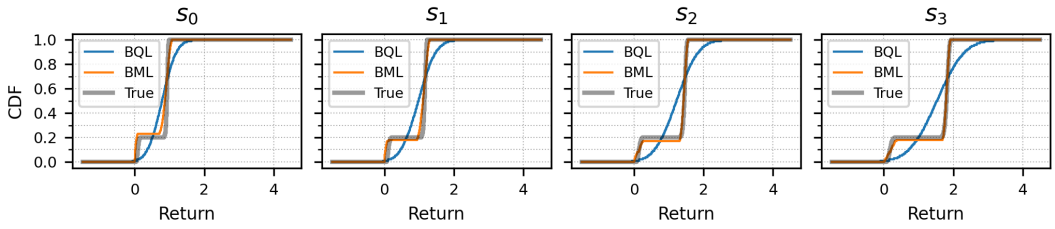


Figure 2: Reconstructed return distributions shown as CDFs in the chain environment.

4.2 Control

We now turn to the control setting. To crystallize the contribution of epistemic uncertainty, we compare BML to Sketch-TD. We chose Sketch-TD as our baseline because it has the same representational capacity but lacks an explicit representation of the epistemic uncertainty. We did not include non-Bayesian baselines for comparison within uncertainty estimators since this lies beyond the scope of our claims. Moreover, since Sketch-TD does not maintain epistemic uncertainty, we pair it with a standard exploration mechanism (ϵ-greedy) in all control experiments.

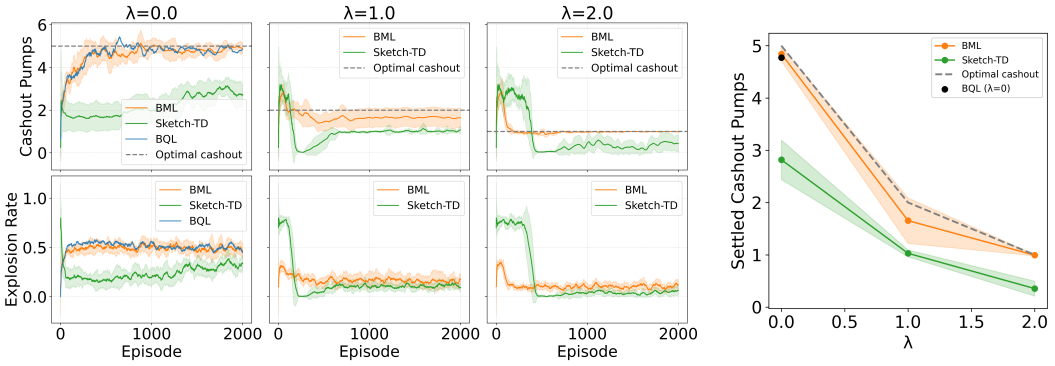
4.2.1 BART

We consider a simple variant of the Balloon Analogue Risk Task (BART) (Lejuez et al., 2002), which poses an explicit sequential risk-taking trade-off. Each episode begins in state $k = 0$, where $k \in \{0, \ldots, N-1\}$ denotes the number of successful pumps so far. The agent has two actions: `PUMP` and `CASH`. Taking `PUMP` either advances to $k+1$ or terminates the episode with an explosion (`BOOM`). The explosion hazard increases with k, where $\mathbb{P}(\texttt{BOOM} \mid k, \texttt{PUMP}) = \frac{1}{N-k}$. So risk grows as the agent pumps more. Taking `CASH` ends the episode deterministically and yields reward proportional to the number of successful pumps accumulated, $R(k, \texttt{CASH}) = k \cdot r$, while all other transitions

yield reward 0. In our experiments we use $N = 10$, $r = 1$, and $\gamma = 1$, train for 2,000 episodes, and average results over several independent runs.

Figure 3 summarizes the BART control results for $\lambda \in \{0, 1, 2\}$. For each λ, we optimize method-specific hyperparameters by grid search (see Supplementary Section 11.2.1). Since standard BQL is risk-neutral, we include it only for $\lambda = 0$. With the risk-neutral objective ($\lambda = 0$) BQL and BML perform similarly well, displaying efficient exploration, which is in stark contrast to Sketch-TD.

For the risk-sensitive objectives ($\lambda > 0$) the learning trajectories in Figure 3a clearly show superior performance of BML. In particular, BML is able to discover near-optimal policies much earlier than Sketch-TD, and with a drastically lower explosion rate. Sketch-TD is insufficiently risk-averse early in learning when (the not represented) epistemic uncertainty should be high; it is also exceedingly risk-averse late in learning, presumably having been discouraged by the early explosions. BML, by contrast, exhibits appropriately cautious exploration and reaches near-optimal behavior earlier.



(a) Learning Dynamics (b) Settled risk-taking

Figure 3: **(a)** Learning dynamics (columns $\lambda \in \{0, 1, 2\}$): 50-episode rolling means of cashout pumps (top; conditioned on cashout) and explosion rate (bottom); dashed gray marks the optimal cashout threshold under $m_1 - \lambda\sigma$. **(b)** Settled cashout vs. λ (final 500 episodes; conditioned on cashout) with optimal curve (dashed gray).

5 Discussion

We introduced *Bayesian Moment Learning* (BML), a model-free TD Bayesian DRL algorithm which combines Bellman-closed mean-embedding sketches (Wenliang et al., 2023) with the Bayesian updating and VPI-style exploration of Bayesian Q-learning (Dearden et al., 1998). BML maintains a posterior over finitely many moments of return distributions. Although its belief does not strictly respect the geometry of the embedding space, our sanity checks show that the issue is transient and benign (Supplementary Section 12); when $v < 0$ early in learning we use $\text{sign}(v)\sqrt{|v|}$ to keep Ψ real and avoid undue pessimism. We focused on two implications of the representation offered by BML: first, its representational capacity substantially exceeds BQL's Gaussian return model. We showed that BML learns better approximations to complex return distributions compared to BQL.

The second implication is for control. We isolated the role of Bayesian uncertainty in risk-sensitive exploration by comparing BML's performance to a non-Bayesian baseline, Sketch-TD (Wenliang et al., 2023), with the same representational capacity. In BART, BML's rsVPI exploration policy learned faster and suffered substantially fewer catastrophic outcomes, allowing it to converge closer to the optimal risk-sensitive policy for various degrees of risk aversion. In Supplementary Section 13.2 we additionally show the desired sensitivity of rsVPI to higher-order statistics of return distributions. This highlights the importance of a proper accounting of epistemic uncertainty in control problems where risk cannot be neglected.

This also positions BML within recent work at the interface of distributional and Bayesian RL. Standard DRL primarily represents aleatoric return uncertainty, while many epistemic RL methods represent uncertainty over expected values. A closely related exception is Luis et al. (2024), which takes a model-based Bayesian perspective and learns an epistemic distribution over value functions induced by model uncertainty. BML takes a complementary, model-free route: its posterior lives over Bellman-consistent moment embeddings of the return distribution, so epistemic uncertainty is attached to risk-relevant return structure itself and can be used directly by rsVPI for risk-aware exploration.

An important extension to our work is to consider risk utilities beyond moments that exhibit favourable theoretical properties, such as conditional value-at-risk (CVaR) (Rockafellar & Uryasev, 2002). While BML can explore CVaR via imputed distributions, alternative embeddings may be more efficient. For instance, Wenliang et al. (2023) report that bounded translation features (e.g., sigmoid/Gaussian) and bin sketches can improve numerical stability and admit theoretical guarantees, but they forgo exact Bellman closure and require tracking an (estimated) return range. It remains a promising avenue for future research to examine risk-sensitive exploration with such approximate sketches. Our experiments deliberately isolate one such mechanism in interpretable settings; extensions to higher-dimensional and continuous-control domains in combination with offline RL also merit attention.

References

Georgy Antonov and Peter Dayan. Exploring uncertainty in distributional reinforcement learning. In *Reinforcement Learning Conference*, 2024.

Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. In *International conference on machine learning*, pp. 449–458. Pmlr, 2017.

Marc G Bellemare, Will Dabney, and Mark Rowland. *Distributional reinforcement learning*. MIT Press, 2023.

Will Dabney, Mark Rowland, Marc Bellemare, and Rémi Munos. Distributional reinforcement learning with quantile regression. In *Proceedings of the AAAI conference on artificial intelligence*, volume 32, 2018.

Richard Dearden, Nir Friedman, Stuart Russell, et al. Bayesian q-learning. *Aaai/iaai*, 1998:761–768, 1998.

Matthias Heger. Consideration of risk in reinforcement learning. In *Machine Learning Proceedings 1994*, pp. 105–111. Elsevier, 1994.

Ronald A Howard. Information value theory. *IEEE Transactions on systems science and cybernetics*, 2(1):22–26, 1966.

Carl W Lejuez, Jennifer P Read, Christopher W Kahler, Jerry B Richards, Susan E Ramsey, Gregory L Stuart, David R Strong, and Richard A Brown. Evaluation of a behavioral measure of risk taking: the balloon analogue risk task (bart). *Journal of Experimental Psychology: Applied*, 8(2): 75, 2002.

Carlos E. Luis, Alessandro G. Bottero, Julia Vinogradska, Felix Berkenkamp, and Jan Peters. Value-distributional model-based reinforcement learning. *Journal of Machine Learning Research*, 25 (298):1–42, 2024.

Shie Mannor and John Tsitsiklis. Mean-variance optimization in markov decision processes. *arXiv preprint arXiv:1104.5601*, 2011.

Krikamol Muandet, Kenji Fukumizu, Bharath Sriperumbudur, and Bernhard Schölkopf. Kernel mean embedding of distributions: A review and beyond. *Foundations and TrendsÂ® in Machine Learning*, 10(1-2):1–141, 2017.

Will Penny. Bayesian inference for the multivariate normal. *Wellcome Trust Center for Neuroimaging: University College, London*, 2014.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

R Tyrrell Rockafellar and Stanislav Uryasev. Conditional value-at-risk for general loss distributions. *Journal of banking & finance*, 26(7):1443–1471, 2002.

Mark Rowland, Robert Dadashi, Saurabh Kumar, Rémi Munos, Marc G Bellemare, and Will Dabney. Statistics and samples in distributional reinforcement learning. In *International Conference on Machine Learning*, pp. 5528–5536. PMLR, 2019.

Maneesh Sahani and Peter Dayan. Doubly distributional population codes: simultaneous representation of uncertainty and multiplicity. *Neural computation*, 15(10):2255–2279, 2003.

Konrad Schmüdgen et al. *The moment problem*, volume 9. Springer, 2017.

Jonathan So. Estimating the normal-inverse-wishart distribution. *arXiv preprint arXiv:2405.16088*, 2024.

Richard S Sutton, Andrew G Barto, et al. *Reinforcement learning: An introduction*, volume 1. MIT press Cambridge, 1998.

Christopher JCH Watkins and Peter Dayan. Q-learning. *Machine learning*, 8(3):279–292, 1992.

Li Kevin Wenliang, Grégoire Delétang, Matthew Aitchison, Marcus Hutter, Anian Ruoss, Arthur Gretton, and Mark Rowland. Distributional Bellman operators over mean embeddings. *arXiv preprint arXiv:2312.07358*, 2023.

Supplementary Materials

The following content was not necessarily subject to peer review.

6 Bellman-closed polynomial sketches with and without a constant feature

This section resolves a notational point behind the polynomial sketch Bellman operator used in the main text (Section 3.1). With an *augmented* polynomial feature map that includes a constant term, polynomial sketches satisfy an exact *linear* Bellman-closedness relation. In the main text we instead use the *reduced* map $\boldsymbol{\phi}(z) = (z, z^2, \ldots, z^k)^\top$, which drops the constant feature (since its embedding coordinate is deterministically 1), yielding an equivalent *affine* recursion of the form $\boldsymbol{\phi}(r + \gamma z) = B_r \boldsymbol{\phi}(z) + \boldsymbol{\phi}(r)$.

6.1 Augmented map and exact linear closure

Define the augmented polynomial features $\tilde{\boldsymbol{\phi}}(z) = (1, z, z^2, \ldots, z^k)^\top \in \mathbb{R}^{k+1}$. By the binomial theorem, for each reward $r \in \mathbb{R}$ there exists a matrix $\tilde{B}_r \in \mathbb{R}^{(k+1)\times(k+1)}$ such that

$$\tilde{\boldsymbol{\phi}}(r + \gamma z) = \tilde{B}_r \, \tilde{\boldsymbol{\phi}}(z) \qquad \text{for all } z \in \mathbb{R}. \tag{7}$$

Indexing monomial orders by $i, j \in \{0, \ldots, k\}$, one convenient closed form is

$$(\tilde{B}_r)_{ij} = \begin{cases} \binom{i}{j} \, r^{\,i-j} \, \gamma^{\,j}, & j \le i, \\ 0, & j > i. \end{cases} \tag{8}$$

6.2 Reduced map and induced affine closure

In our Bayesian model we use the reduced features $\boldsymbol{\phi}(z) = (z, z^2, \ldots, z^k)^\top \in \mathbb{R}^k$, so that $\tilde{\boldsymbol{\phi}}(z) = (1, \boldsymbol{\phi}(z)^\top)^\top$. Partition $\tilde{B}_r$ conformably as

$$\tilde{B}_r = \begin{pmatrix} 1 & \mathbf{0}^\top \\ \boldsymbol{\phi}(r) & B_r \end{pmatrix}, \tag{9}$$

where $B_r \in \mathbb{R}^{k\times k}$. Extracting the lower block of $\tilde{\boldsymbol{\phi}}(r + \gamma z) = \tilde{B}_r \tilde{\boldsymbol{\phi}}(z)$ yields the reduced-space affine recursion

$$\boldsymbol{\phi}(r + \gamma z) = B_r \, \boldsymbol{\phi}(z) + \boldsymbol{\phi}(r), \tag{10}$$

which is the form used in Section 3.

Example ($k = 2$). With $\boldsymbol{\phi}(z) = (z, z^2)^\top$, the induced matrices and offset are

$$B_r = \begin{pmatrix} \gamma & 0 \\ 2r\gamma & \gamma^2 \end{pmatrix}, \qquad \boldsymbol{\phi}(r) = \begin{pmatrix} r \\ r^2 \end{pmatrix}, \tag{11}$$

so Eq. 10 holds exactly.

In the mixture posterior updates (Section 3.2.1 and Supplementary Section 7.7), the pseudo-observation takes the affine form

$$\mathbf{x} = \boldsymbol{\phi}(r + \gamma Z(s', a')) = B_r \, \mathbf{y} + \boldsymbol{\phi}(r), \qquad \text{where } \mathbf{y} = \boldsymbol{\phi}(Z(s', a')). \tag{12}$$

Dropping the constant feature avoids carrying a deterministic coordinate in the Normal–Wishart belief, while preserving exact Bellman-closedness via the induced affine recursion above.

7 Normal–Wishart identities

Notation and mapping to the main text. In this appendix we use the standard Normal–Wishart notation $(\boldsymbol{\mu}, \Lambda)$ for clarity. In the main text, the role of $\boldsymbol{\mu}$ is played by the embedding mean $\mathbf{U}(s, a)$, and $\mathbf{x}$ denotes an (actual or pseudo) observation in embedding space. Throughout, d is the dimension of $\mathbf{x}$ (or $\mathbf{U}$), which in our setting equals the number of reduced moment features k (Section 6).

7.1 Prior

We use the Normal–Wishart prior

$$(\boldsymbol{\mu}, \Lambda) \sim \mathrm{NW}(\boldsymbol{\mu}_0, \lambda, W, \nu), \tag{13}$$

where $\boldsymbol{\mu}_0 \in \mathbb{R}^d$, $\lambda > 0$, $W \in \mathbb{R}^{d\times d}$ is positive definite, and $\nu > d - 1$. This means

$$\Lambda \sim \mathcal{W}(W, \nu), \qquad \boldsymbol{\mu} \mid \Lambda \sim \mathcal{N}\big(\boldsymbol{\mu}_0, (\lambda\Lambda)^{-1}\big). \tag{14}$$

(Here $\mathcal{W}(W, \nu)$ denotes the Wishart distribution on the *precision* matrix Λ with scale matrix W.)

7.2 Posterior

Given i.i.d. data $\mathbf{x}_{1:n}$ with

$$\mathbf{x}_i \sim \mathcal{N}(\boldsymbol{\mu}, \Lambda^{-1}), \tag{15}$$

the posterior is also Normal–Wishart:

$$(\boldsymbol{\mu}, \Lambda) \mid \mathbf{x}_{1:n} \sim \mathrm{NW}(\boldsymbol{\mu}_n, \lambda_n, W_n, \nu_n), \tag{16}$$

with updates

$$\lambda_n = \lambda + n, \qquad \boldsymbol{\mu}_n = \frac{\lambda\boldsymbol{\mu}_0 + n\bar{\mathbf{x}}}{\lambda + n}, \qquad \nu_n = \nu + n, \tag{17}$$

and

$$W_n^{-1} = W^{-1} + \sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top + \frac{n\lambda}{n + \lambda}(\bar{\mathbf{x}} - \boldsymbol{\mu}_0)(\bar{\mathbf{x}} - \boldsymbol{\mu}_0)^\top, \tag{18}$$

where $\bar{\mathbf{x}} = \frac{1}{n}\sum_{i=1}^{n}\mathbf{x}_i$.

One-sample (rank-one) update and efficient inversion. For the special case $n = 1$, the sample mean is $\bar{\mathbf{x}} = \mathbf{x}_1$ and the scatter term $\sum_{i=1}^{n}(\mathbf{x}_i - \bar{\mathbf{x}})(\mathbf{x}_i - \bar{\mathbf{x}})^\top$ vanishes. Defining

$$\mathbf{z} := \mathbf{x}_1 - \boldsymbol{\mu}_0, \qquad c := \frac{\lambda}{\lambda + 1},$$

the Normal–Wishart posterior hyperparameters reduce to

$$\lambda_1 = \lambda + 1, \qquad \boldsymbol{\mu}_1 = \frac{\lambda\boldsymbol{\mu}_0 + \mathbf{x}_1}{\lambda + 1}, \qquad \nu_1 = \nu + 1, \qquad W_1^{-1} = W^{-1} + c\,\mathbf{z}\mathbf{z}^\top. \tag{19}$$

Equation (19) shows that W^{-1} is updated by a *rank-one* outer product. As a result, $W_1 = (W^{-1} + c\,\mathbf{z}\mathbf{z}^\top)^{-1}$ can be computed efficiently via the Woodbury matrix identity (equivalently, the Sherman–Morrison rank-one update). In general, for invertible A and C,

$$(A + UCV)^{-1} = A^{-1} - A^{-1}U\big(C^{-1} + VA^{-1}U\big)^{-1}VA^{-1}. \tag{20}$$

With $A = W^{-1}$, $U = \mathbf{z}$, $\mathbf{V} = \mathbf{z}^\top$, and $C = c$ (a scalar), we obtain

$$W_1 = (W^{-1} + c\,\mathbf{z}\mathbf{z}^\top)^{-1} = W - W\mathbf{z}\big(c^{-1} + \mathbf{z}^\top W\mathbf{z}\big)^{-1}\mathbf{z}^\top W, \tag{21}$$

where the term $\big(c^{-1} + \mathbf{z}^\top W\mathbf{z}\big)$ is scalar.

7.3 Marginal of $\boldsymbol{\mu}$

The marginal distribution of $\boldsymbol{\mu}$ under the Normal–Wishart prior is a multivariate Student-t:

$$\boldsymbol{\mu} \sim t_{\nu-d+1}\left(\boldsymbol{\mu}_0,\ \frac{1}{\lambda(\nu-d+1)}\,W^{-1}\right). \tag{22}$$

The mean exists when $\nu > d$ (equivalently $\nu - d + 1 > 1$). The covariance exists when $\nu > d + 1$ (equivalently $\nu - d + 1 > 2$), and is

$$\mathrm{Cov}[\boldsymbol{\mu}] = \frac{\nu-d+1}{\nu-d-1}\cdot\frac{1}{\lambda(\nu-d+1)}\,W^{-1} = \frac{1}{\lambda(\nu-d-1)}\,W^{-1}. \tag{23}$$

7.4 Prior predictive

Assume

$$p(\mathbf{x} \mid \boldsymbol{\mu}, \Lambda) = \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \Lambda^{-1}), \qquad (\boldsymbol{\mu}, \Lambda) \sim \mathrm{NW}(\boldsymbol{\mu}_0, \lambda, W, \nu). \tag{24}$$

Integrating out $\boldsymbol{\mu}$ conditional on Λ gives

$$\begin{aligned} p(\mathbf{x} \mid \Lambda) &= \int_{\mathbb{R}^d} p(\mathbf{x} \mid \boldsymbol{\mu}, \Lambda)\, p(\boldsymbol{\mu} \mid \Lambda)\, d\boldsymbol{\mu} \\ &= \int_{\mathbb{R}^d} \mathcal{N}(\mathbf{x}; \boldsymbol{\mu}, \Lambda^{-1})\, \mathcal{N}\big(\boldsymbol{\mu}; \boldsymbol{\mu}_0, (\lambda\Lambda)^{-1}\big)\, d\boldsymbol{\mu} \\ &= \mathcal{N}\big(\mathbf{x}; \boldsymbol{\mu}_0, (1+\lambda^{-1})\Lambda^{-1}\big) = \mathcal{N}\left(\mathbf{x}; \boldsymbol{\mu}_0, \left(\frac{\lambda}{\lambda+1}\right)^{-1}\Lambda^{-1}\right). \end{aligned} \tag{25}$$

Integrating out Λ yields a multivariate Student-t prior predictive (Penny, 2014):

$$\mathbf{x}_{\text{new}} \sim t_{\nu-d+1}\left(\boldsymbol{\mu}_0,\ \frac{\lambda+1}{\lambda(\nu-d+1)}\,W^{-1}\right). \tag{26}$$

It is sometimes convenient to also record the corresponding *inverse-scale / precision* matrix for this t:

$$\left(\frac{\lambda+1}{\lambda(\nu-d+1)}\,W^{-1}\right)^{-1} = \frac{\lambda}{\lambda+1}(\nu-d+1)\,W. \tag{27}$$

The predictive covariance (when $\nu > d + 1$) is

$$\mathrm{Cov}[\mathbf{x}_{\text{new}}] = \frac{\lambda+1}{\lambda(\nu-d-1)}\,W^{-1}. \tag{28}$$

7.5 Posterior predictive

Using the posterior hyperparameters $(\boldsymbol{\mu}_n, \lambda_n, W_n, \nu_n)$ from Section 7.2, the posterior predictive is (Penny, 2014)

$$\mathbf{x}_{\text{new}} \mid \mathbf{x}_{1:n} \sim t_{\nu_n-d+1}\left(\boldsymbol{\mu}_n,\ \frac{\lambda_n+1}{\lambda_n(\nu_n-d+1)}\,W_n^{-1}\right), \tag{29}$$

with corresponding inverse-scale / precision matrix

$$\left(\frac{\lambda_n+1}{\lambda_n(\nu_n-d+1)}\,W_n^{-1}\right)^{-1} = \frac{\lambda_n}{\lambda_n+1}(\nu_n-d+1)\,W_n. \tag{30}$$

7.6 Sufficient statistics and expectations (for KL projection)

The Wishart and inverse-Wishart densities (as used for reference/parameterization) are

$$\mathcal{W}(\Lambda; W, \nu) = \frac{|\Lambda|^{(\nu-d-1)/2}\exp\{-\frac{1}{2}\mathrm{tr}(W^{-1}\Lambda)\}}{2^{\nu d/2}\,|W|^{\nu/2}\,\Gamma_d\left(\frac{\nu}{2}\right)}, \tag{31}$$

and

$$\mathcal{IW}(\Lambda; W, \nu) = \frac{|W|^{\nu/2} \exp\{-\frac{1}{2}\mathrm{tr}(W\Lambda^{-1})\}}{2^{\nu d/2}\,|\Lambda|^{(\nu+d+1)/2}\,\Gamma_d(\frac{\nu}{2})}. \tag{32}$$

(With this convention, $\mathcal{W}(\Lambda; W, \nu) = \mathcal{IW}(\Lambda^{-1}; W^{-1}, \nu)$.)

According to So (2024), one convenient sufficient-statistic vector for the Normal–inverse-Wishart (NIW) can be written (up to constant rescalings) as

$$T(\mathbf{U}, \Lambda) = \left(-\tfrac{1}{2}\log|\Lambda|, -\tfrac{1}{2}\mathrm{vec}(\Lambda^{-1}), \Lambda^{-1}\mathbf{U}, -\tfrac{1}{2}\mathbf{U}^\top\Lambda^{-1}\mathbf{U}\right). \tag{33}$$

Replacing Λ^{-1} by Λ yields a corresponding sufficient-statistic set for the Normal–Wishart (NW); up to equivalent constant rescalings this gives

$$T(\mathbf{U}, \Lambda) = \left(\log|\Lambda|,\ \Lambda,\ \Lambda\mathbf{U},\ \mathbf{U}^\top\Lambda\mathbf{U}\right), \tag{34}$$

which matches the sufficient statistics used in the main text for forward-KL projection.

We also record the corresponding expectations under $(\mathbf{U}, \Lambda) \sim \mathrm{NW}(\boldsymbol{\mu}_0, \lambda, W, \nu)$.

Log-determinant.

$$\mathbb{E}[\log|\Lambda|] = \sum_{i=1}^{d} \psi\left(\frac{\nu+1-i}{2}\right) + d\log 2 + \log|W|. \tag{35}$$

where $\psi(\cdot)$ denotes the digamma function.

First moment.

$$\mathbb{E}[\Lambda] = \nu W. \tag{36}$$

Mixed moment. Using iterated expectation,

$$\mathbb{E}[\Lambda\mathbf{U}] = \mathbb{E}\big[\Lambda\,\mathbb{E}[\mathbf{U} \mid \Lambda]\big] = \mathbb{E}[\Lambda\,\boldsymbol{\mu}_0] = (\nu W)\boldsymbol{\mu}_0. \tag{37}$$

Quadratic form. Using $\mathbb{E}[\mathbf{X}^\top A\mathbf{X}] = \mathbf{m}^\top A\mathbf{m} + \mathrm{tr}(A\Sigma)$ for $\mathbf{X} \sim \mathcal{N}(\mathbf{m}, \Sigma)$,

$$\begin{aligned}
\mathbb{E}[\mathbf{U}^\top\Lambda\mathbf{U}] &= \mathbb{E}\big[\mathbb{E}[\mathbf{U}^\top\Lambda\mathbf{U} \mid \Lambda]\big] \\
\mathbb{E}[\mathbf{U}^\top\Lambda\mathbf{U} \mid \Lambda] &= \boldsymbol{\mu}_0^\top\Lambda\boldsymbol{\mu}_0 + \mathrm{tr}\big(\Lambda(\lambda\Lambda)^{-1}\big) = \boldsymbol{\mu}_0^\top\Lambda\boldsymbol{\mu}_0 + \frac{d}{\lambda}, \\
\Rightarrow\quad \mathbb{E}[\mathbf{U}^\top\Lambda\mathbf{U}] &= \boldsymbol{\mu}_0^\top(\nu W)\boldsymbol{\mu}_0 + \frac{d}{\lambda}.
\end{aligned} \tag{38}$$

7.7 Normal–Wishart Projection via Forward KL

The mixture posterior $p_{\mathrm{mix}}(\mathbf{U}, \Lambda)$ from Section 3.2.1 is not, in general, Normal–Wishart. Following Bayesian Q-learning, we approximate it by a Normal–Wishart distribution via forward KL projection. We find

$$q^\star = \arg\min_{q \in NW} \mathrm{KL}(p_{\mathrm{mix}} \,\|\, q)\,, \tag{39}$$

such that $q(\mathbf{U}, \Lambda) = \mathrm{NW}(\boldsymbol{\mu}, \lambda, W, \nu)$. Since the Normal–Wishart is an exponential family, this projection is equivalent to matching expectations of sufficient statistics under p_{mix} and q.

We use the sufficient statistics outlined in (34) and define the target moments

$$M_1 = \mathbb{E}_{p_{\mathrm{mix}}}[\log|\Lambda|], \quad M_2 = \mathbb{E}_{p_{\mathrm{mix}}}[\Lambda], \quad \mathbf{M}_3 = \mathbb{E}_{p_{\mathrm{mix}}}[\Lambda\mathbf{U}], \quad M_4 = \mathbb{E}_{p_{\mathrm{mix}}}[\mathbf{U}^\top\Lambda\mathbf{U}] \tag{40}$$

We estimate $M_1, M_2, \mathbf{M}_3, M_4$ by Monte Carlo over the conditional posteriors induced by the sampled bootstrap targets. Specifically, we sample $\mathbf{y}^{(i)} \sim p_{\mathbf{Y}'}$ and set $\mathbf{x}^{(i)} = B_r\mathbf{y}^{(i)} + \boldsymbol{\phi}(r)$. For each $\mathbf{x}^{(i)}$, conjugacy yields a conditional Normal–Wishart posterior

$$p(\mathbf{U}, \Lambda \mid \mathbf{x}^{(i)}) = \mathrm{NW}(\boldsymbol{\mu}^{(i)}, \lambda^{(i)}, W^{(i)}, \nu^{(i)}), \tag{41}$$

from which $\mathbb{E}[\log|\Lambda|]$, $\mathbb{E}[\Lambda]$, $\mathbb{E}[\Lambda\mathbf{U}]$, and $\mathbb{E}[\mathbf{U}^\top\Lambda\mathbf{U}]$ are available in closed form. Averaging these conditional expectations over $i = 1, \dots, K$ produces estimates of $M_1, M_2, \mathbf{M}_3, M_4$.

Using the Normal–Wishart moment identities for $\mathbb{E}[\log|\Lambda|]$, $\mathbb{E}[\Lambda]$, $\mathbb{E}[\Lambda\mathbf{U}]$, and $\mathbb{E}[\mathbf{U}^\top\Lambda\mathbf{U}]$ (Supplementary Section 7.6), moment matching yields the closed-form relations

$$\boldsymbol{\mu} = M_2^{-1}\mathbf{M}_3, \tag{42}$$

$$\lambda = \frac{d}{M_4 - \boldsymbol{\mu}^\top M_2 \boldsymbol{\mu}}, \qquad \text{requiring } M_4 > \boldsymbol{\mu}^\top M_2\boldsymbol{\mu}, \tag{43}$$

$$W = \frac{M_2}{\nu}, \qquad \text{with } M_2 \text{ symmetric positive definite.} \tag{44}$$

The remaining parameter $\nu > d - 1$ is obtained by solving the scalar equation implied by matching $M_1 = \mathbb{E}[\log|\Lambda|]$:

$$\psi_d(\nu) + d\log 2 + \log|M_2| - d\log\nu = M_1, \tag{45}$$

where $\psi_d(\nu) := \sum_{j=1}^{d} \psi\big(\frac{\nu+1-j}{2}\big)$ is the multivariate digamma. We solve (45) numerically and then set $W = M_2/\nu$.

8 Normal–Gamma identities (BQL)

Model and parameterization. Bayesian Q-learning (BQL) assumes a Gaussian return model with unknown mean and precision:

$$z \mid Q, \tau \sim \mathcal{N}(Q, \tau^{-1}), \tag{46}$$

together with a conjugate Normal–Gamma (NG) prior,

$$(Q, \tau) \sim \mathrm{NG}(\mu_0, \lambda, \alpha, \beta), \tag{47}$$

with the *rate* parameterization

$$\tau \sim \mathrm{Gamma}(\alpha, \beta), \qquad Q \mid \tau \sim \mathcal{N}\big(\mu_0, (\lambda\tau)^{-1}\big). \tag{48}$$

Here μ_0 is the prior mean of Q, $\lambda > 0$ scales the conditional variance $\mathrm{Var}(Q \mid \tau) = 1/(\lambda\tau)$, and $\alpha, \beta > 0$ control the Gamma prior over τ.

Marginal of Q. Marginalizing τ yields a Student-t distribution,

$$Q \sim t_{2\alpha}\left(\mu_0,\ \frac{\beta}{\alpha\lambda}\right), \tag{49}$$

whose variance exists when $\alpha > 1$ and equals

$$\mathrm{Var}(Q) = \frac{\beta}{\lambda(\alpha - 1)}. \tag{50}$$

Prior predictive for a return sample. Marginalizing (Q, τ) yields the prior predictive

$$z_{\text{new}} \sim t_{2\alpha}\left(\mu_0,\ \frac{\beta(\lambda+1)}{\alpha\lambda}\right), \tag{51}$$

with variance (for $\alpha > 1$)

$$\mathrm{Var}(z_{\text{new}}) = \frac{\beta(\lambda+1)}{\lambda(\alpha-1)}. \tag{52}$$

Posterior updates. Given i.i.d. data $x_{1:n}$ with $x_i \sim \mathcal{N}(Q, \tau^{-1})$, the posterior is

$$(Q, \tau) \mid x_{1:n} \sim \mathrm{NG}(\mu_n, \lambda_n, \alpha_n, \beta_n), \tag{53}$$

where

$$\lambda_n = \lambda + n, \qquad \mu_n = \frac{\lambda\mu_0 + n\bar{x}}{\lambda + n},$$

$$\alpha_n = \alpha + \frac{n}{2}, \qquad \beta_n = \beta + \frac{1}{2}\sum_{i=1}^{n}(x_i - \bar{x})^2 + \frac{n\lambda}{2(\lambda+n)}(\bar{x} - \mu_0)^2, \tag{54}$$

and $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$. For $n = 1$, letting $z = x_1 - \mu_0$ and $c = \frac{\lambda}{\lambda+1}$,

$$\lambda_1 = \lambda + 1, \quad \mu_1 = \frac{\lambda\mu_0 + x_1}{\lambda + 1}, \quad \alpha_1 = \alpha + \frac{1}{2}, \quad \beta_1 = \beta + \frac{1}{2}\, c\, z^2. \tag{55}$$

The marginal posterior over Q is

$$Q \mid x_{1:n} \sim t_{2\alpha_n}\left(\mu_n,\ \frac{\beta_n}{\alpha_n \lambda_n}\right). \tag{56}$$

9 Additional Theoretical Details

9.1 Equivalence of first-moment updates for BQL and BML

On terminal (non-bootstrapped) transitions, BML receives a direct embedding observation $\mathbf{x} = \boldsymbol{\phi}(r)$ and updates its Normal–Wishart posterior by the i.i.d. rule in Section 7.2. BQL receives the corresponding scalar observation $x_0 = [\mathbf{x}]_0$ and updates its Normal–Gamma posterior by Section 8.

Claim. Under the parameter mapping

$$\mu_0^{\mathrm{NG}} = [\boldsymbol{\mu}_0]_0, \qquad \lambda^{\mathrm{NG}} = \lambda, \qquad \alpha = \frac{\nu - d + 1}{2}, \qquad \beta = \frac{[W^{-1}]_{00}}{2}, \tag{57}$$

the induced posterior marginal over Q in BQL equals the posterior marginal over the first coordinate $[\mathbf{U}]_0$ in BML after any number of direct observations.

Proof (one-step; extension by induction). Consider one direct observation $\mathbf{x}$ and its first coordinate $x_0 = [\mathbf{x}]_0$. From the NW one-sample update (Eq. 19) with $\mathbf{z} = \mathbf{x} - \boldsymbol{\mu}_0$ and $c = \lambda/(\lambda + 1)$, we have

$$[W_1^{-1}]_{00} = [W^{-1}]_{00} + c\, z_0^2, \qquad z_0 = x_0 - [\boldsymbol{\mu}_0]_0. \tag{58}$$

From the NG one-sample update (Eq. 55) we have

$$\beta_1 = \beta + \frac{1}{2}\, c\, (x_0 - \mu_0^{\mathrm{NG}})^2. \tag{59}$$

Under (57), $\mu_0^{\mathrm{NG}} = [\boldsymbol{\mu}_0]_0$ and $\beta = \frac{1}{2}[W^{-1}]_{00}$, so (58) and (59) imply

$$\beta_1 = \frac{1}{2}[W_1^{-1}]_{00}. \tag{60}$$

Next, the NG marginal posterior is Student-t (Eq. 56) with degrees of freedom $2\alpha_1$. The NW marginal posterior over the mean is Student-t by applying Eq. 22 to the posterior hyperparameters, hence the first coordinate has degrees of freedom $\nu_1 - d + 1$.

Finally, with $\alpha = (\nu - d + 1)/2$ we obtain $2\alpha_1 = 2(\alpha + \frac{1}{2}) = \nu_1 - d + 1$, and with (60) the Student-t scale parameters agree as well. Since the posterior locations also match via $\mu_0^{\mathrm{NG}} = [\boldsymbol{\mu}_0]_0$ and $\lambda^{\mathrm{NG}} = \lambda$, the two univariate Student-t posteriors coincide. □

9.2 Monte Carlo estimation of rsVPI

We estimate $\mathbb{E}[\Psi(s,a)]$ and $\mathrm{rsVPI}(s,a)$ from L posterior samples $\mathbf{U}^{(\ell)}(s,a)$ by computing $\psi_a^{(\ell)} = \rho(\mathbf{U}^{(\ell)}(s,a))$ and using sample averages $\widehat{\mathbb{E}}[\Psi(a)] = \frac{1}{L}\sum_{\ell=1}^{L} \psi_a^{(\ell)}$ and

$$\widehat{\mathrm{rsVPI}}(s,a) = \begin{cases} \frac{1}{L}\sum_{\ell=1}^{L} \left(\widehat{\mathbb{E}}[\Psi(a_2)] - \psi_a^{(\ell)}\right)_+, & a = a_1, \\ \frac{1}{L}\sum_{\ell=1}^{L} \left(\psi_a^{(\ell)} - \widehat{\mathbb{E}}[\Psi(a_1)]\right)_+, & a \neq a_1. \end{cases} \tag{61}$$

We do not reject posterior samples during action selection, since hard filtering can bias early beliefs and suppress exploration. Moment-feasibility checks are used only as an empirical diagnostic in Section 4.

Algorithm 1 BML control with rsVPI

Require: Normal–Wishart beliefs $\{\mathcal{B}_{s,a}\}$, discount γ, risk functional Ψ, sample counts L, K

for each interaction step at state s **do**
  for each action $a \in \mathcal{A}(s)$ **do**
    Draw $\mathbf{U}^{(1)}(s,a), \dots, \mathbf{U}^{(L)}(s,a)$ from the marginal Student-t belief induced by $\mathcal{B}_{s,a}$
    Compute risk scores $\psi_a^{(\ell)} = \Psi(\mathbf{U}^{(\ell)}(s,a))$, for $\ell = 1, \dots, L$
    Estimate $\widehat{\mathbb{E}}[\Psi(s,a)]$ and $\widehat{\mathrm{rsVPI}}(s,a)$ from these samples
  end for
  Select $a \in \arg\max_b \{\widehat{\mathbb{E}}[\Psi(s,b)] + \widehat{\mathrm{rsVPI}}(s,b)\}$
  Execute a, observe (r, s')
  if s' is terminal **then**
    Set $\mathbf{x} = \boldsymbol{\phi}(r)$
    Update $\mathcal{B}_{s,a}$ by the conjugate Normal–Wishart posterior with observation $\mathbf{x}$
  else
    Select bootstrap distribution $q(\cdot \mid s')$
    if control **then**
      $a^\star \in \arg\max_b \widehat{\mathbb{E}}[\Psi(s',b)]$
      $q(\cdot \mid s') \leftarrow \delta_{a^\star}$
    else
      $q(\cdot \mid s') \leftarrow \pi(\cdot \mid s')$
    end if
    for $i = 1, \dots, K$ **do**
      Draw $a_i' \sim q(\cdot \mid s')$
      Draw $\mathbf{y}^{(i)}$ from the posterior predictive distribution induced by $\mathcal{B}_{s',a_i'}$
      Form pseudo-observation $\mathbf{x}^{(i)} = B_r \mathbf{y}^{(i)} + \boldsymbol{\phi}(r)$
      Form the conditional Normal–Wishart posterior for $\mathcal{B}_{s,a}$ given $\mathbf{x}^{(i)}$
    end for
    Match the mixture expectations $M_1, M_2, \mathbf{M}_3, M_4$ of these conditional posteriors
    Project the mixture back to the Normal–Wishart family by forward KL, and store the result as $\mathcal{B}_{s,a}$
  end if
  Recompute the rsVPI policy at s using the updated belief
end for

10 Algorithmic Summary of BML

For completeness, we summarize the BML control loop in Algorithm 1, tying together rsVPI action selection and the per-transition Bayesian update. We maintain a Normal–Wishart belief $\mathcal{B}_{s,a} = \mathrm{NW}(\boldsymbol{\mu}_{s,a}, \lambda_{s,a}, W_{s,a}, \nu_{s,a})$ over the mean embedding $\mathbf{U}(s,a)$ at every state–action pair, using the reduced polynomial map $\boldsymbol{\phi}(z) = (z, \dots, z^k)^\top$ (Section 3.1). Priors are set as in Supplementary Section 11.1.1, and $B_r, \boldsymbol{\phi}(r)$ are the Bellman-closed affine maps of Section 3.1.

In policy evaluation, the bootstrap distribution $q(\cdot \mid s')$ is the fixed evaluation policy $\pi(\cdot \mid s')$, and no policy-improvement step is performed. In control, rsVPI is used for action selection at the current state, while the bootstrapped target uses the greedy action under the current expected risk score.

11 Experimental details and reproducibility

11.1 Policy evaluation

11.1.1 Prior matching between BQL and BML

In the policy-evaluation experiments we initialize BQL (Normal–Gamma) and BML (Normal–Wishart) so that they encode the same epistemic uncertainty about the *mean return*. For polynomial embeddings $\boldsymbol{\phi}(z) = (z, z^2, \dots)^\top$, the first coordinate of the embedding mean is the first raw moment, $[\mathbf{U}]_0 = \mathbb{E}[Z]$, which plays the same role as BQL's mean parameter Q.

Matching the marginal over the mean. BQL uses the Normal–Gamma prior $(Q, \tau) \sim \mathrm{NG}(\mu_0^{\mathrm{NG}}, \lambda^{\mathrm{NG}}, \alpha, \beta)$ (Section 8), whose marginal over Q is Student-t:

$$Q \sim t_{2\alpha}\left(\mu_0^{\mathrm{NG}},\ \frac{\beta}{\alpha\lambda^{\mathrm{NG}}}\right), \tag{62}$$

see (49). BML uses the Normal–Wishart prior $(\mathbf{U}, \Lambda) \sim \mathrm{NW}(\boldsymbol{\mu}_0, \lambda, W, \nu)$ (Section 7), whose marginal over $\mathbf{U}$ is multivariate Student-t:

$$\mathbf{U} \sim t_{\nu-d+1}\left(\boldsymbol{\mu}_0,\ \frac{1}{\lambda(\nu-d+1)}\, W^{-1}\right), \tag{63}$$

see (22) (Section 7.3). Hence the first coordinate satisfies

$$[\mathbf{U}]_0 \sim t_{\nu-d+1}\left([\boldsymbol{\mu}_0]_0,\ \frac{1}{\lambda(\nu-d+1)}\, [W^{-1}]_{00}\right). \tag{64}$$

To make the priors over Q and $[\mathbf{U}]_0$ identical (same location, degrees of freedom, and scale), we set

$$\mu_0^{\mathrm{NG}} = [\boldsymbol{\mu}_0]_0, \qquad \lambda^{\mathrm{NG}} = \lambda, \qquad \alpha = \frac{\nu-d+1}{2}, \qquad \beta = \frac{[W^{-1}]_{00}}{2}. \tag{65}$$

This is the same NG–NW mapping used in Section 9.1 to show equivalence of terminal first-moment updates.

Initializing higher-moment coordinates for BML. Matching the mean coordinate does not determine the remaining entries of the NW prior mean vector $\boldsymbol{\mu}_0 \in \mathbb{R}^d$. In policy evaluation we initialize these entries using the prior predictive return distribution implied by BQL's Normal–Gamma model.

Specifically, marginalizing (Q, τ) yields the prior predictive over a return sample z (Eq. 51):

$$z_{\mathrm{new}} \sim t_{2\alpha}\left(\mu_0^{\mathrm{NG}},\ \frac{\beta(\lambda+1)}{\alpha\lambda}\right). \tag{66}$$

We set the BML prior mean embedding to the corresponding raw moments of this predictive distribution:

$$[\boldsymbol{\mu}_0]_0 = \mu_0^{\mathrm{NG}}, \tag{67}$$

$$[\boldsymbol{\mu}_0]_j = \mathbb{E}\left[z_{\mathrm{new}}^{j+1}\right], \qquad j = 1, \dots, d-1. \tag{68}$$

These moments exist provided $2\alpha > j+1$, and under our hyperparameter choices below this condition holds for all required coordinates.

Hyperparameter choices used in our experiments. We set $\lambda^{\mathrm{NG}} = \lambda = 1$ and $\mu_0^{\mathrm{NG}} = 0$. We choose ν to satisfy two requirements: (i) the NW marginal covariance exists, requiring $\nu > d+1$

(Section 7.3); and (ii) the predictive Student-t in Eq. 66 has finite raw moments up to order d, requiring $2\alpha > d$, i.e.,

$$\nu - d + 1 > d \quad \Longleftrightarrow \quad \nu > 2d - 1, \tag{69}$$

since $\alpha = (\nu - d + 1)/2$ under Eq. 65. In practice we use

$$\nu = \max(2d,\ d + 2), \qquad \text{and hence} \qquad \alpha = \frac{\nu - d + 1}{2}. \tag{70}$$

Finally, we choose an isotropic Wishart scale $W = wI$, which implies $[W^{-1}]_{00} = 1/w$ and therefore (by Eq. 65)

$$\beta = \frac{1}{2w}. \tag{71}$$

This single scalar w controls the prior uncertainty in embedding space. Using the NW covariance formula (Section 7.3),

$$\mathrm{Cov}(\mathbf{U}) = \frac{1}{\lambda(\nu - d - 1)}\, W^{-1} = \frac{1}{\lambda(\nu - d - 1)w}\, I. \tag{72}$$

Equivalently, to target $\mathrm{Cov}(\mathbf{U}) = \sigma_0^2 I$, we set

$$w = \frac{1}{\lambda(\nu - d - 1)\sigma_0^2}. \tag{73}$$

11.1.2 Chain Environment

The Chain environment used in Section 4.1 is a deterministic episodic MDP with five states $\mathcal{S} = \{s_0, s_1, s_2, s_3, s_4\}$, start state s_0, terminal state s_4, and discount factor $\gamma = 0.8$. We use a single action, so the dynamics form a simple chain:

$$s_0 \to s_1 \to s_2 \to s_3 \to s_4, \tag{74}$$

and the episode terminates upon entering s_4.

Rewards are 0 on all nonterminal transitions. The only stochastic reward is obtained on the transition into the terminal state: if $s = s_3$ and $s' = s_4$, then

$$R \sim 2 \cdot \begin{cases} \mathrm{Beta}(10, 100), & \text{with probability } 0.2, \\ \mathrm{Beta}(100, 10), & \text{with probability } 0.8, \end{cases} \tag{75}$$

and otherwise $R \equiv 0$.

11.1.3 Moment absolute relative error

In Figure 1 (Section 4.1.1) and Figures 6–9 (Section 13.1.1), the y-axis reports the *absolute relative error* of each raw return moment. Let $m_i^\star$ denote the true ith raw moment of the return distribution at the relevant state, and let $\hat{m}_i$ denote the corresponding estimate produced by the algorithm at a given episode. We plot

$$\mathrm{err}_i \ = \ \left| \frac{m_i^\star - \hat{m}_i}{m_i^\star} \right|. \tag{76}$$

(For the chain environment used in these figures, $m_i^\star > 0$ for the moments considered, so the denominator is nonzero.)

11.2 Control

11.2.1 BART hyperparameter search

We evaluate control in the BART environment with $N = 10$ and actions $\{\texttt{PUMP}, \texttt{CASH}\}$. Each episode begins in state $k = 0$ (number of successful pumps so far). Choosing CASH terminates the episode deterministically and yields reward k (we use reward-per-pump $r = 1$). Choosing PUMP transitions either to BOOM (terminal) with probability $1/(N-k)$ or to $k+1$ otherwise, with reward 0 on all non-CASH transitions. We report results for Ψ defined in (5) with $\lambda \in \{0, 1, 2\}$.

Tuning objective. Hyperparameters were selected by grid search on a held-out set of random seeds using the same risk-sensitive objective used in the experiments, computed from realized episode outcomes. For episode e, define the realized return

$$G_e = \begin{cases} k_e, & \text{if the episode ends by CASH at pump count } k_e, \\ 0, & \text{if the episode ends by BOOM.} \end{cases} \tag{77}$$

Given a window of episodes, define empirical moments $\hat{m}_1 = \frac{1}{M}\sum_e G_e$ and $\hat{m}_2 = \frac{1}{M}\sum_e G_e^2$, and compute the final-window score

$$\widehat{J}_\lambda = \hat{m}_1 - \lambda\,\sigma(\hat{v}), \tag{78}$$

$$\hat{v} = \hat{m}_2 - \hat{m}_1^2, \tag{79}$$

$$\sigma(\hat{v}) = \mathrm{sign}(\hat{v})\sqrt{|\hat{v}|}. \tag{80}$$

For $\lambda = 0$, this reduces to maximizing the empirical mean return.

Protocol and seed splits. For each candidate hyperparameter setting and each $\lambda \in \{0, 1, 2\}$, we trained the corresponding agent for 2000 episodes and evaluated $\widehat{J}_\lambda$ on the final 500 episodes (the "final-window" objective). Each candidate was evaluated over multiple independent random seeds and the scores were averaged. We used different tuning budgets per method to reflect computational cost: BML used 5 tuning seeds, Sketch-TD used 10 tuning seeds, and BQL used 10 tuning seeds. After selecting hyperparameters, we ran the final reported curves using a disjoint set of 10 new random seeds (separate from all tuning seeds), again training for 2000 episodes per run.

What is tuned. We tuned method-specific hyperparameters separately for each λ. For Sketch-TD we tuned the learning rate and epsilon-greedy exploration rate (α, ϵ). For BML we tuned a single scalar prior scale controlling the initial epistemic uncertainty $\mathrm{Cov}(\mathbf{U}) = \sigma_0^2 I$. For BQL we tuned an analogous scalar prior scale controlling the Normal–Gamma prior, and we evaluate BQL only for $\lambda = 0$ (risk-neutral baseline).

Search grids. For each $\lambda \in \{0, 1, 2\}$, we performed a grid search over:

$$\text{Sketch-TD:}\quad \alpha \in \{0.01, 0.05, 0.1, 0.2\}, \qquad \epsilon \in \{0.01, 0.05, 0.1, 0.2\}, \tag{81}$$

$$\text{BML:}\quad \sigma_0^2 \in \{500, 1000, 5000, 10000, 30000, 50000, 100000\}. \tag{82}$$

For BQL (tuned and evaluated only at $\lambda = 0$), we searched the same scalar prior scale values:

$$\text{BQL:}\quad \sigma_0^2 \in \{500, 1000, 5000, 10000, 30000, 50000, 100000\}. \tag{83}$$

For BML and BQL, σ_0^2 is a scalar controlling the magnitude of the initial prior uncertainty (diagonal scale) and therefore the strength of early exploration induced by posterior uncertainty.

Action selection during tuning. For BML, we used rsVPI and for Sketch-TD we used epsilon-greedy control with respect to its learned value estimates and the specified risk objective through the target definition. BQL used its standard VPI rule over expected return.

Optimal cashout reference. In Figure 3 we additionally plot the analytically optimal cashout threshold for each λ, computed by enumerating fixed-threshold policies ("pump to x, then cash") and selecting the maximizer of $\mathbb{E}[G] - \lambda \operatorname{Std}(G)$ using the environment's closed-form statistics. These optimal thresholds are used only for visualization and interpretation, not for hyperparameter selection.

Plotting. Learning curves are shown as rolling averages with window size 50 episodes for both cashout level (conditioned on `CASH`) and explosion rate. "Settled cashout" is computed as the mean cashout level over the final 500 episodes.

12 Moment-feasibility diagnostics

This section evaluates the practical impact of the modeling approximation introduced in Section 3.2: our posterior over the embedding mean $\mathbf{U}(s, a) \in \mathbb{R}^k$ does not explicitly enforce the structural constraints that define valid embeddings. We therefore report diagnostics tracking how the posterior evolves relative to the feasible set during learning. To ensure these trends are not artifacts of initialization, we present results under two priors: (i) an uninformative baseline with $\boldsymbol{\mu}_0 = \mathbf{0}$, and (ii) a BQL-matched initialization that sets $\boldsymbol{\mu}_0$ using moments of the BQL-implied prior predictive (Section 11.1.1). We use the same chain environment as in Section 4.1, with full specifications given in Section 11.1.2.

12.1 Geometry of Valid Embeddings.

Just as in BQL, the Gaussian likelihood in (2) is an approximation; however, because we place this likelihood on the embedded return $\mathbf{Y} = \boldsymbol{\phi}(Z)$ and maintain uncertainty over its mean $\mathbf{U} = \mathbb{E}[\boldsymbol{\phi}(Z)]$, we can capture complex (e.g., multimodal or heavy-tailed) return distributions through the chosen embedding without committing to a restrictive parametric form for Z itself. However, since $\boldsymbol{\phi} : \mathbb{R} \to \mathbb{R}^k$, the embedded return lies in the one-dimensional subset $\mathcal{M} := \boldsymbol{\phi}(\mathbb{R}) \subset \mathbb{R}^k$. Thus, the mean embedding $\mathbf{U} = \mathbb{E}[\mathbf{Y}]$ cannot be an arbitrary point in $\mathbb{R}^k$ but has to lie in the convex hull of $\mathcal{M}$, such that $\mathbf{U} \in \operatorname{conv}(\mathcal{M})$, since $\mathbf{U}$ is an expectation of points on $\mathcal{M}$.

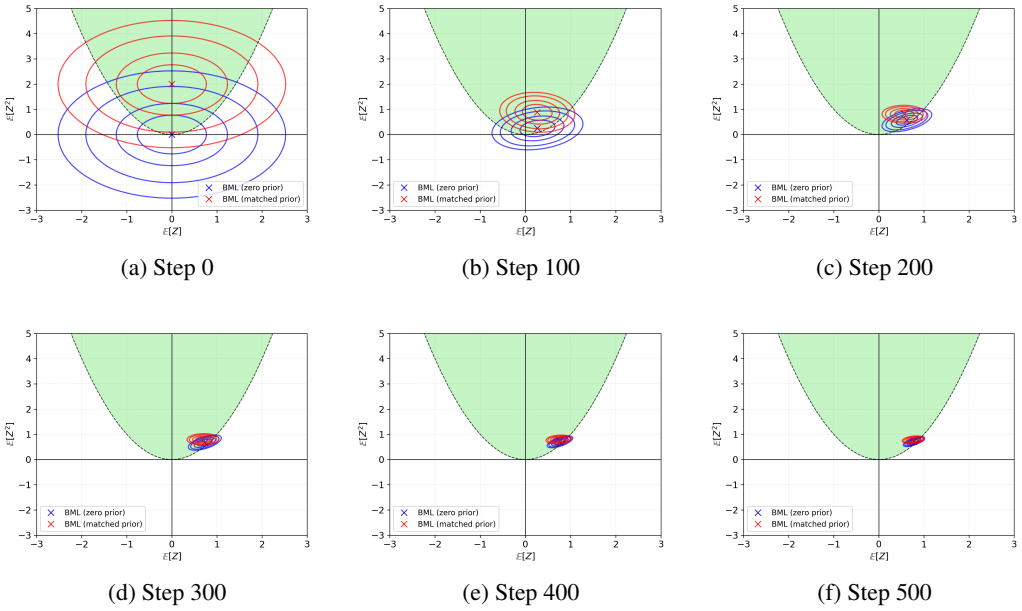
For polynomial feature maps, this corresponds to the usual feasibility constraints on finite-order moment vectors. For random variables supported on $\mathbb{R}$, this reduces to the truncated Hamburger moment problem, which imposes positive-semidefiniteness (and related extension) conditions on the associated Hankel moment matrices (Schmüdgen et al., 2017). Our Gaussian model does not enforce these constraints explicitly, so the posterior may assign density to moment vectors that are not realizable by any distribution over Z. We therefore evaluate this effect empirically by tracking how often posterior samples correspond to moment-feasible vectors (Section 12.3), where we observe that the fraction of infeasible samples decreases over training. The aforementioned moment-feasibility check is used only as a diagnostic and is not enforced during learning.

12.2 Contour plots (two-moment case).

This constraint is easiest to visualize when $m = 2$ and $\mathbf{U} = (m_1, m_2)$ encodes the first two raw moments. If $\boldsymbol{\phi}(z) = (z, z^2)$, then $\boldsymbol{\phi}(z)$ lies on the curve $\mathcal{M} = \{(x, x^2) : x \in \mathbb{R}\}$. Since $\mathbf{U} = \mathbb{E}[\boldsymbol{\phi}(Z)]$ is an expectation of points on $\mathcal{M}$, it must lie in the convex hull of $\mathcal{M}$, which in this case is the epigraph of the parabola $m_2 \geq m_1^2$. Figure 4 shades this feasible region (light green) above the parabola.

Figure 4 shows the evolution of the BML posterior over $\mathbf{U}(s_0, a)$ for the start state s_0 in the chain environment. Early in training, the Gaussian embedding model assigns nontrivial probability mass outside the feasible set. As experience accumulates and the terminal reward signal propagates backward, the posterior contracts and shifts so that most of its mass concentrates in the moment-feasible region $m_2 \geq m_1^2$. Notably, this occurs at s_0, several steps away from the terminal reward, indicating

that feasibility emerges under repeated bootstrapped updates rather than only from direct terminal observations. This qualitative convergence is observed under both prior initializations.



(a) Step 0 (b) Step 100 (c) Step 200

(d) Step 300 (e) Step 400 (f) Step 500

Figure 4: Evolution of the posterior over $\mathbf{U}(s_0, a)$ in the (m_1, m_2) plane. The feasible region is $m_2 \geq m_1^2$.

12.3 Fraction of feasible samples (higher-order moments).

For higher-dimensional embeddings, visualization is not practical, so we quantify feasibility by sampling from the posterior. At each episode, we draw approximately 10^4 samples of the moment vector $\mathbf{U}(s, a)$ for a fixed state–action pair and test whether each sampled vector is compatible with some probability measure on $\mathbb{R}$.

For $m = 2$, feasibility reduces to the nonnegative variance constraint $m_2 - m_1^2 \geq 0$. For higher-order moment vectors, we apply a truncated Hamburger moment feasibility test, which checks whether the finite moment sequence admits a representing measure on $\mathbb{R}$ (via the standard Hankel matrix positivity and extendability conditions), following Schmüdgen et al. (2017). We observe that the fraction of infeasible samples decreases over training, indicating that the posterior progressively concentrates on the moment-feasible set. Figure 5 shows this trend for the two-moment case and for larger moment dimensions.

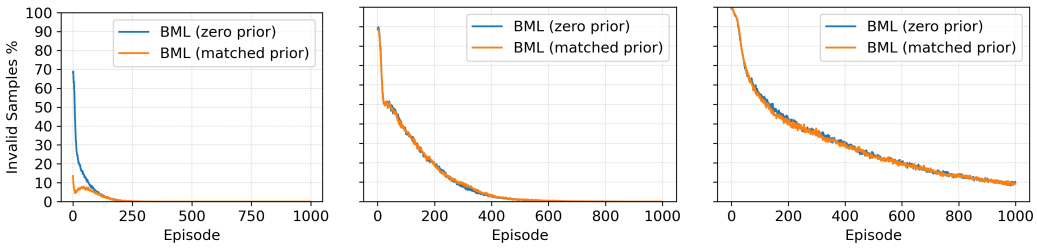
One thing to note is that changing the dimension requires tuning the prior uncertainty. We use an isotropic scale for the NW as outlined in Section 11.1.1 and target $\mathrm{Cov}(\mathbf{U}) = \sigma_0^2 I$ by setting w according to (73). For $d = 3$ we target $\sigma_0^2 = 1$. For $d = 5$, we have $\sigma_0^2 = 0.001$. For $d = 7$ we have $\sigma_0^2 = 0.0001$ and set w consequently.

13 Additional Results

13.1 Policy Evaluation

13.1.1 Additional moment-convergence plots

In Section 4.1.1 we reported moment-convergence results up to the third moment. Here we extend these results to moments up to order seven. We also test robustness to prior initialization. The main-



(a) $d = 3$ (b) $d = 5$ (c) $d = 7$

Figure 5

text plots use the prior-matching procedure in Section 11.1.1 with target covariance $\mathrm{Cov}(\mathbf{U}) = \sigma_0^2 I$ and $\sigma_0^2 = 1$. Below, we repeat the experiment with $\sigma_0^2 \in \{5,\ 0.1,\ 0.01\}$ (Figures 6–9). Across all settings, BML maintains consistently lower error than BQL, and the advantage typically increases with moment order.

13.1.2 Evolution of Imputed Returns Over Training

In Section 4.1.2 we impute a single representative return distribution from BML using the posterior mean embedding $\mathbb{E}[\mathbf{U}(s,a)]$. Here we instead illustrate how the posterior over return distributions evolves during learning. At selected episodes, we sample moment embeddings $\mathbf{U}^{(\ell)}(s,a)$ from the marginal posterior over $\mathbf{U}(s,a)$ (with $k = 10$ moments) and, for each sample, impute a categorical return distribution using the procedure of Wenliang et al. (2023). Figure 10 shows the resulting imputed CDFs at episodes 10, 50, 100, 200, and 500: early in training the imputations are highly dispersed, while over time they contract and become more consistent, reflecting decreasing epistemic uncertainty.

13.1.3 Effect of Embedding Dimension k on Imputation

In Section 4.1.2 we report imputations using a $k = 10$ polynomial embedding. Here we study how the imputed return distribution depends on the embedding dimension k, i.e., the number of raw moments matched. We repeat the same imputation procedure while varying k and holding all other settings fixed. As expected, increasing k typically improves the imputation: matching more moments imposes stronger constraints on the reconstructed distribution, allowing it to better capture higher-order structure that is invisible to low-order moment summaries. Notably, in Figure 11 shows that the improvement largely saturates quickly; already at $k = 5$ the imputed distribution is almost visually indistinguishable from the $k = 10$ imputation.

13.2 Control

13.2.1 K-Paths (Skewness in control)

The K-Paths environment is a finite episodic MDP designed to test the agent's sensitivity to higher moments of the return distribution. There is a distinguished initial state, s_{start}, and K available actions corresponding to K paths. In our setting $K = 3$ with action set $\mathcal{A} = \{a, b, c\}$. Taking action $\chi \in \mathcal{A}$ in s_{start} deterministically transitions the agent to the first state on that path, denoted $\chi 1$.

Each path χ forms a deterministic chain of length L (here $L = 4$) with states $\chi 1, \chi 2, \ldots, \chi L$. From any nonterminal state χi with $i < L$, the transition dynamics are: - If the agent selects the matching action χ, it deterministically transitions to $\chi[i+1]$. - If the agent selects any action $\chi' \neq \chi$, it deterministically transitions to a terminal failure state, s_{end}.

The set of terminal states is $\{aL, bL, cL\} \cup \{s_{\text{end}}\}$. The reward function is sparse. All non-terminal transitions yield reward 0. Transitioning into the failure terminal s_{end} yields a fixed penalty of -10.

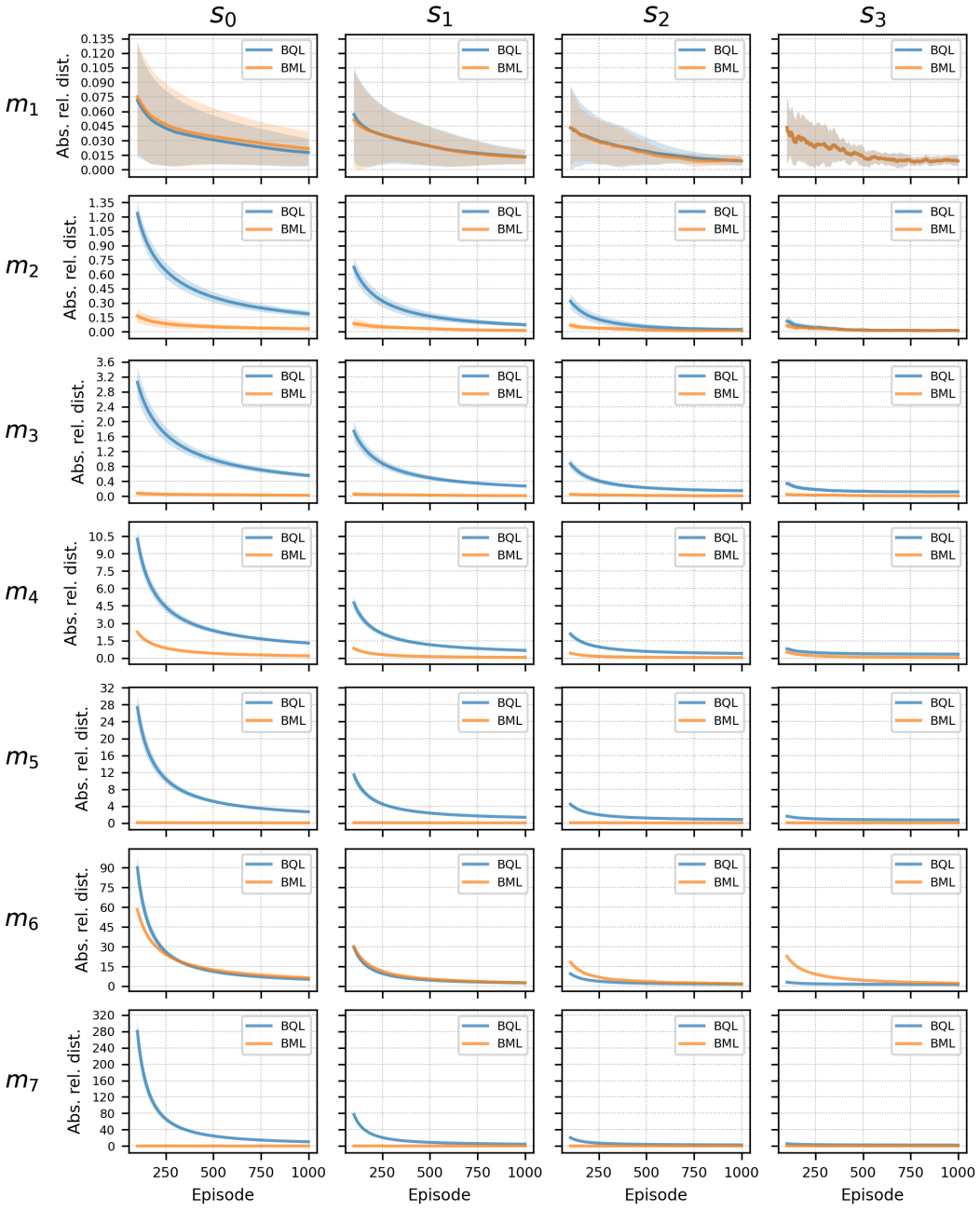


Figure 6: $\sigma_0^2 = 5$

Transitioning into a path's terminal state χL yields a sample from a path-specific terminal reward R_χ that we describe below. The discount factor is $\gamma = 1$. Thus, an optimal agent must select a path at the start and repeatedly choose the same action to reach its corresponding terminal state. Any deviation from the chosen path immediately terminates the episode with a negative reward. Figure 12 depicts the 3-Paths environment used in subsequent experiments.

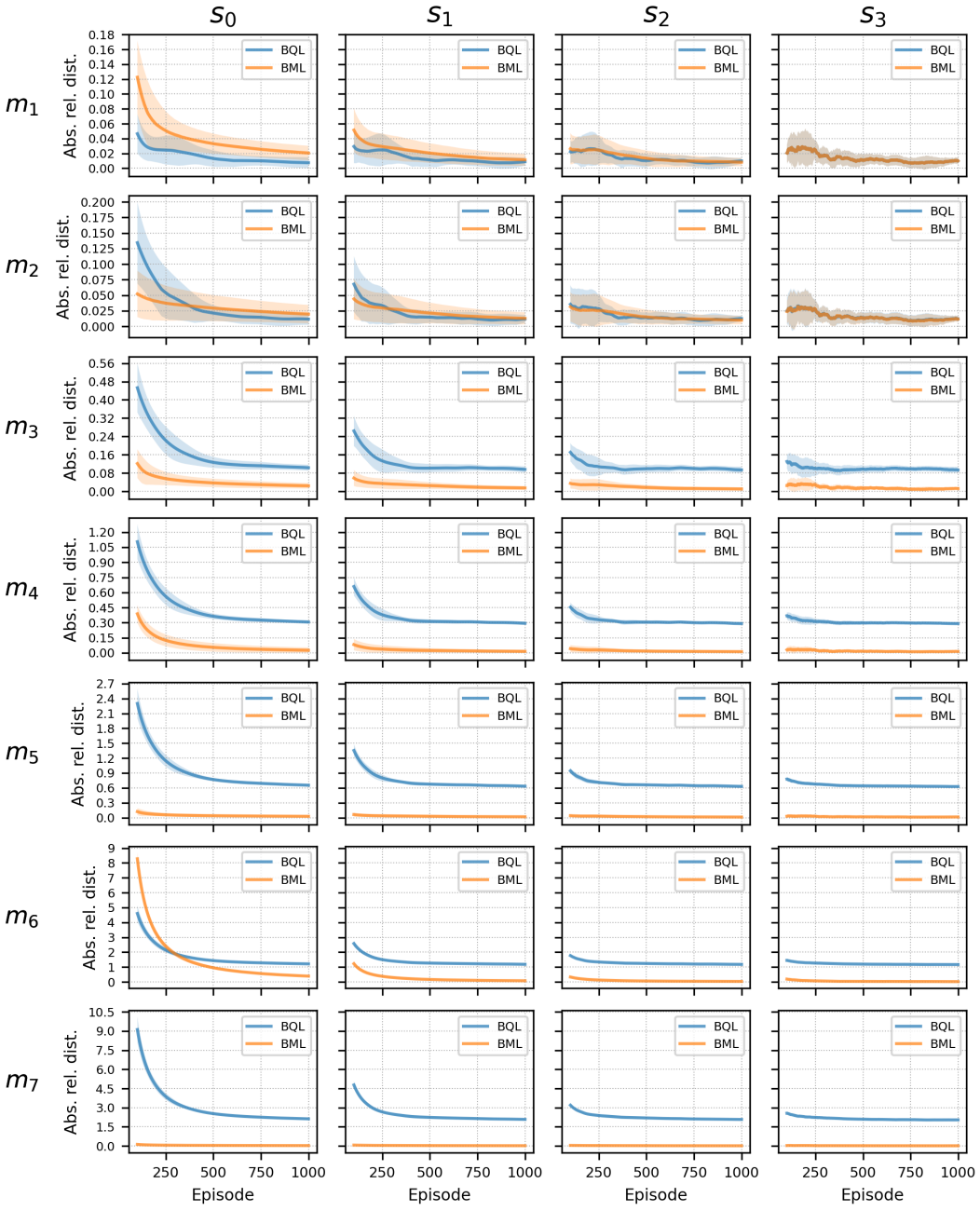


Figure 7: $\sigma_0^2 = 1$

13.2.2 Skew

In the policy-evaluation experiments, BQL matches the second moment of the return distribution reasonably well. However, its estimates deteriorate for higher-order moments, whereas our mean-embedding (BML) posterior remains accurate. To make the practical consequences of this gap explicit, we construct a controlled variant of the Paths environment in which the agent must distinguish options that are identical in mean and variance but differ in skewness.

Specifically, for the three paths A, B, and C the terminal reward distributions are skew-normal, parameterized such that all three paths have the same mean and variance (mean $=$ 2, variance

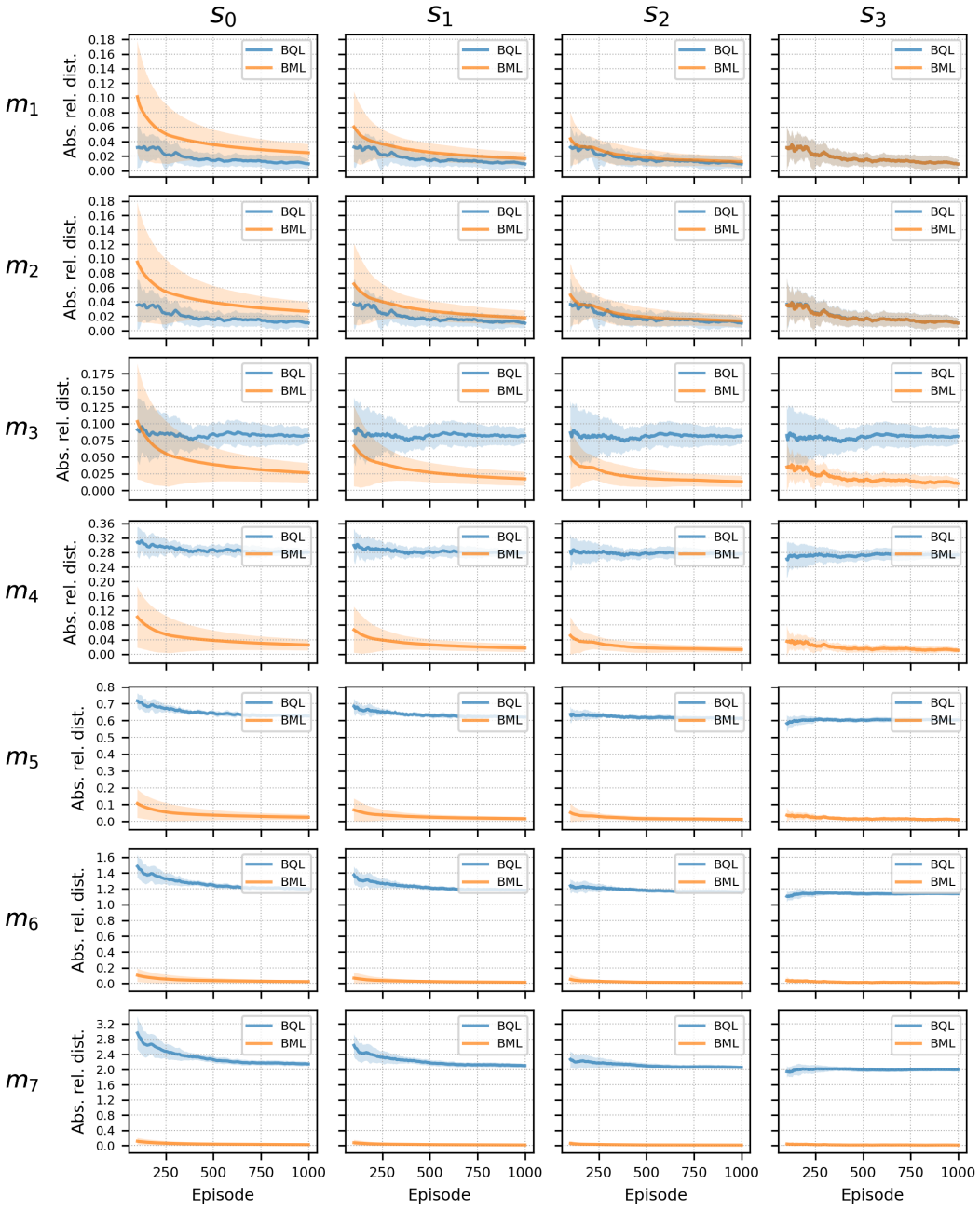


Figure 8: $\sigma_0^2 = 0.1$

matched across paths). Paths A and B are negatively skewed with skewness -0.99, while path C is positively skewed with skewness $+0.99$. Figure 13 shows the reward distributions at the end of the three paths. Because the distributions are indistinguishable at the level of mean and variance, any method that only represents $\mathbb{E}[Z]$ and $\mathbb{E}[Z^2]$ cannot reliably prefer one path over another.

The key distinguishing statistic is skewness (third central moment). We operationalize this by augmenting the risk score with a signed cube-root of the third central moment: $\Psi = m_1 - \lambda\sigma(v) + \alpha\gamma$ where $\gamma = (m_3 - 3m_2m_1 + 2m_1^3)^{1/3}$ and $\sigma(v)$ is the dispersion term defined in (5).

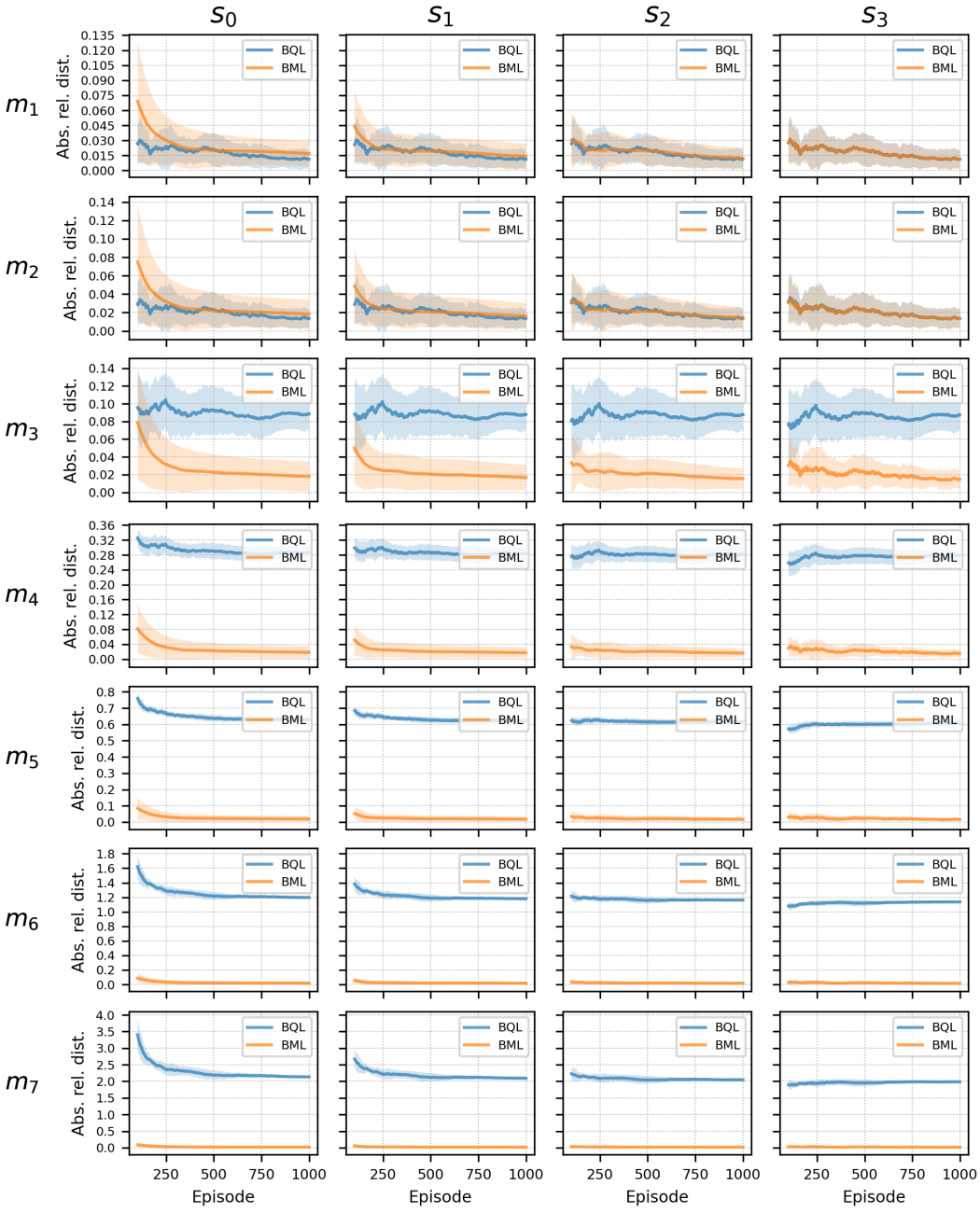


Figure 9: $\sigma_0^2 = 0.01$

The BML agent selects path C (the positively skewed option) far more frequently, while BQL selects among A, B, and C approximately uniformly, reflecting its inability to discriminate between the matched-mean/variance alternatives. Figure 14 reports the mean cumulative action frequencies across 100 runs (shaded regions indicate $\pm$ one standard deviation).

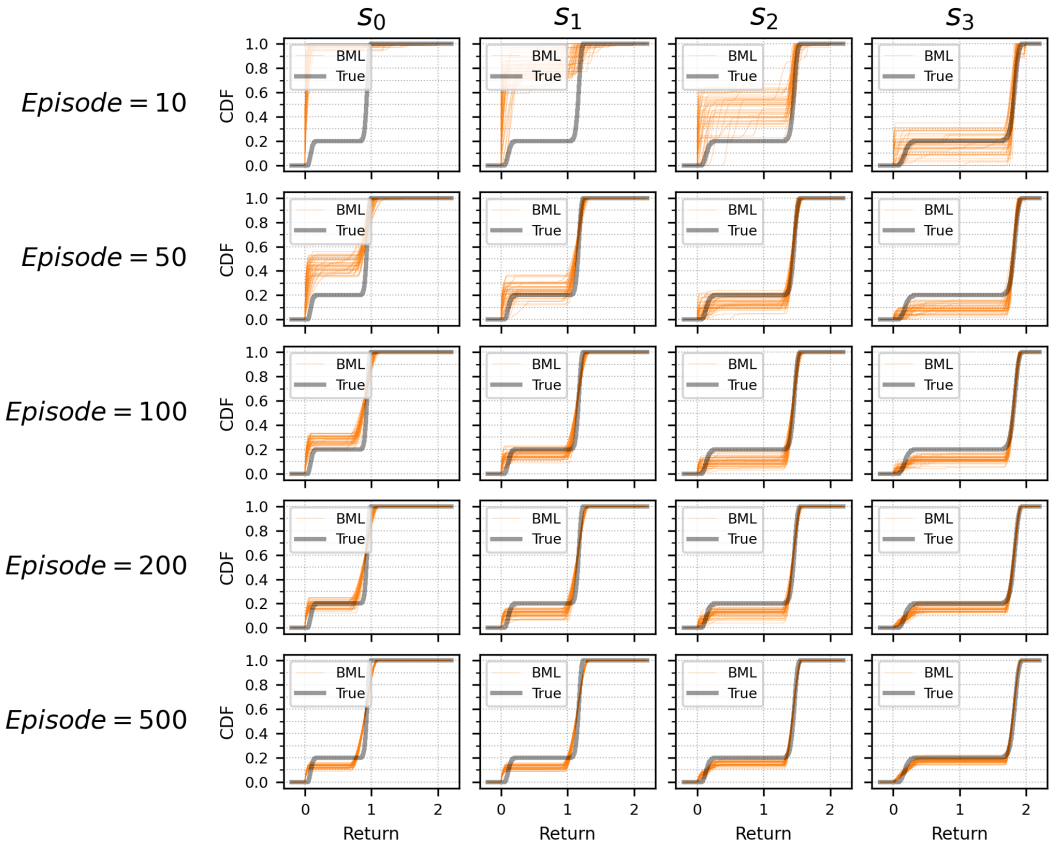


Figure 10: Evolution of posterior-implied imputations over training. At each listed episode we draw samples $\mathbf{U}^{(\ell)}(s, a)$ from the BML marginal posterior (with $k = 10$ moments) and impute a categorical return distribution for each sample; the plotted curves are the corresponding imputed CDFs. The spread across curves visualizes epistemic uncertainty, which shrinks as training progresses.

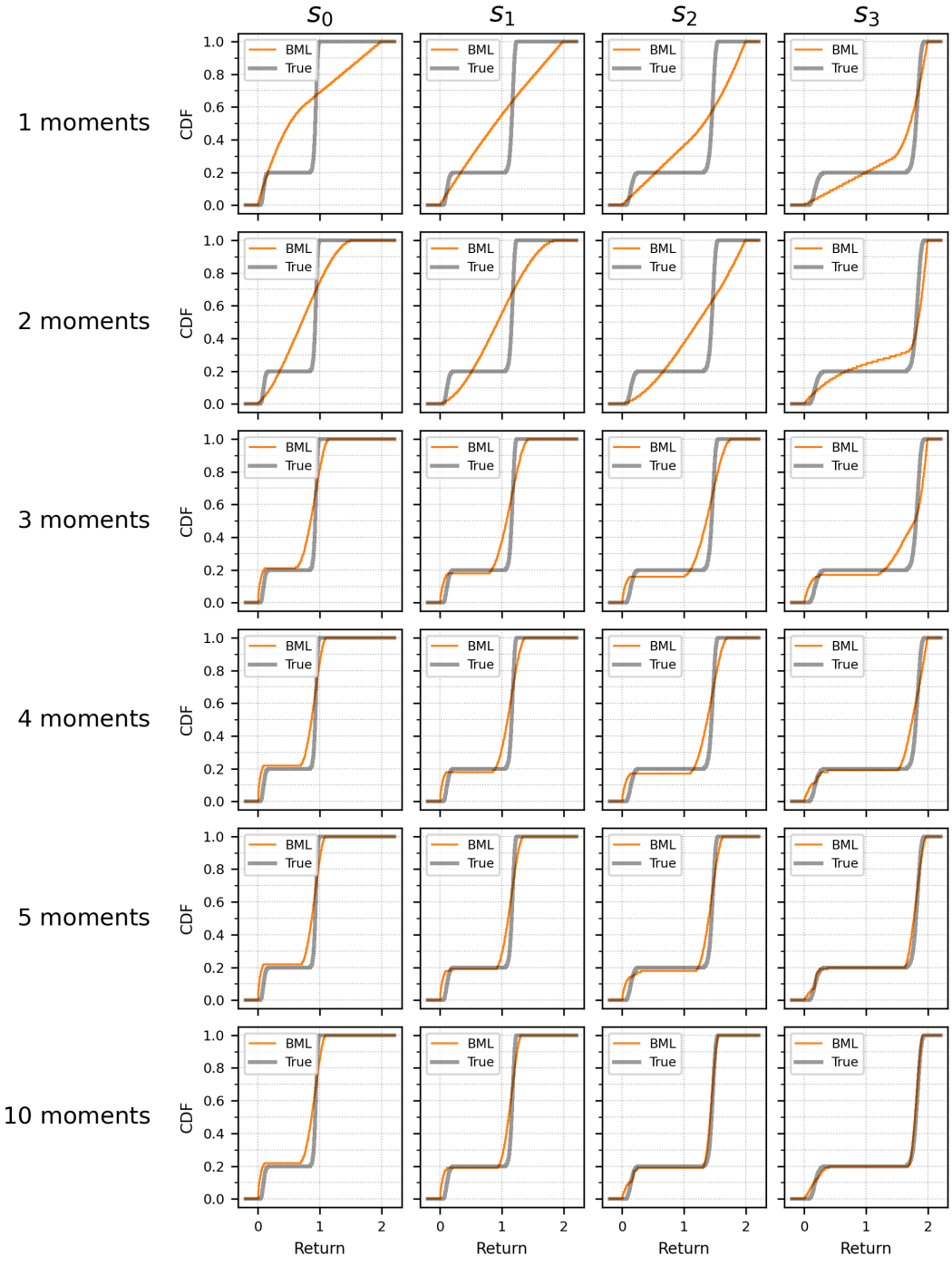


Figure 11: Effect of embedding dimension on imputation. Imputed return CDFs obtained by matching polynomial moment embeddings of increasing dimension k (number of raw moments), with all other settings held fixed. Imputation quality improves rapidly with k and largely saturates by $k = 5$, after which the $k = 5$ and $k = 10$ curves are nearly visually identical.

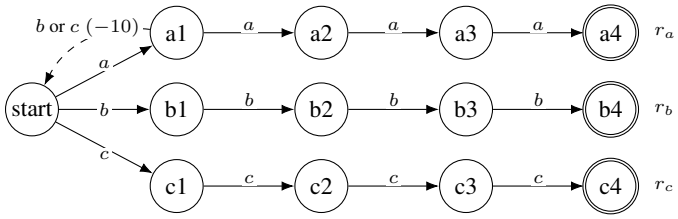


Figure 12: Paths environment (3 paths, chain length 4). Only one incorrect action from $a1$ back to that start state is shown as an example for clarity.

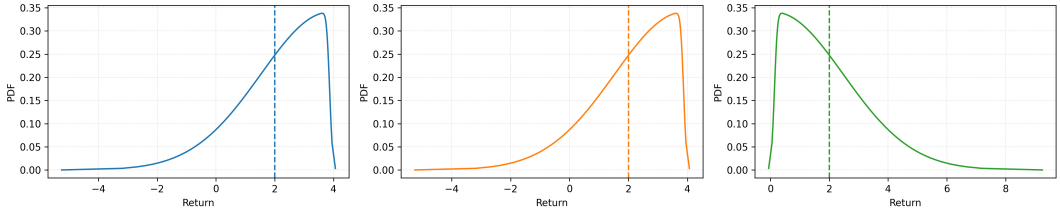
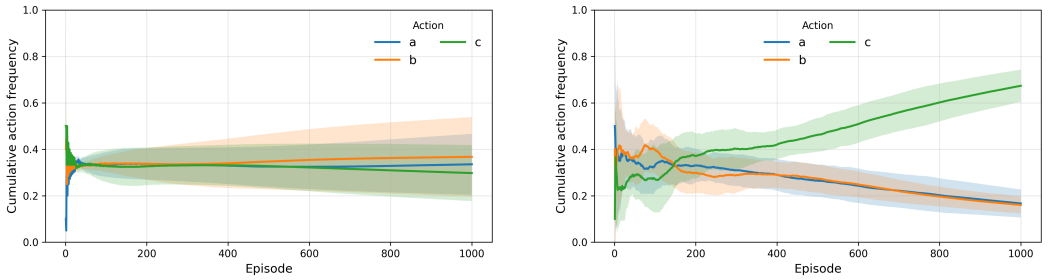


Figure 13: Terminal reward distributions for paths A, B, and C (left to right) in the skewness experiment. All three are matched in mean $= 2$ and variance $= 2$, but A and B are negatively skewed while C is positively skewed, so they differ only in higher-order structure.



(a) BQL (b) BML

Figure 14: Skewness experiment: cumulative action frequencies at the start state over training, averaged over 100 independent runs. The x-axis is episode number and the y-axis is cumulative frequency of choosing each path. BQL remains approximately uniform across A, B, and C, while BML increasingly concentrates on path C (the positively skewed option). Shaded bands show ± 1 standard deviation across runs.