Offline Policy Selection under Uncertainty

Anonymous Author(s)
Affiliation
Address
email

Abstract

The presence of uncertainty in policy evaluation significantly complicates the process of policy ranking and selection in real-world settings. We formally consider *offline policy selection* as learning preferences over a set of policy prospects given a fixed experience dataset. While one can select or rank policies based on point estimates of their expected values or high-confidence intervals, access to the full distribution over one's belief of the policy value enables more flexible selection algorithms under a wider range of downstream evaluation metrics. We propose a Bayesian approach for estimating this belief distribution in terms of posteriors of distribution correction ratios derived from stochastic constraints. Empirically, despite being Bayesian, the credible intervals obtained are competitive with state-of-the-art frequentist approaches in confidence interval estimation. More importantly, we show how the belief distribution may be used to rank policies with respect to arbitrary downstream policy selection metrics, and empirically demonstrate that this selection procedure significantly outperforms existing approaches, such as ranking policies according to mean or high-confidence lower bound value estimates.

1 Introduction

Off-policy evaluation (OPE) [53] in the context of reinforcement learning (RL) is often motivated as a way to mitigate risk in practical applications where deploying a policy might incur significant cost or safety concerns [60]. Indeed, by providing a *point estimate* of the value of a *target policy* solely from a static *offline* dataset of logged experience in the environment, OPE can help practitioners determine whether a target policy is or is not safe and worthwhile to deploy. Still, in many practical applications the ability to accurately estimate the online value of a specific policy is less of a concern than the ability to select or rank a given *set* of policies (one of which may be the currently deployed policy). For example, in recommendation systems, a practitioner may have a large number of policies trained offline using various hyperparameters, while cost and safety constraints only allow a few of those policies to be deployed as live experiments. Which policies should be chosen to form the small subset that will be evaluated online?

This problem, related to but subtly different from OPE, is *offline policy selection* [17, 51, 36]. The original motivations for OPE were arguably with offline policy selection in mind [53, 28], the idea being that one can use estimates of the value of a set of policies to rank and then select from this set. Accordingly, there is a rich literature of approaches for computing point estimates of the value of the policy [19, 4, 31, 59, 45, 69, 62, 32, 66], as well as estimating high-confidence lower and upper bounds on a target policy's value [60, 36, 4, 25, 22, 11, 34].

These existing OPE approaches may be readily applied to the recommendation systems example above by using either mean or high-confidence bounds estimates on each candidate policy to rank the set and picking the top few to deploy online. However, such a naïve approach ignores crucial differences between the OPE problem setting and the downstream evaluation criteria a practitioner prioritizes. For example, when choosing a few policies out of a large number of policies, a recommendation systems practitioner may have a number of objectives in mind: They may strive to ensure that the

Submitted to 35th Conference on Neural Information Processing Systems (NeurIPS 2021). Do not distribute.

policy with the overall highest groundtruth value is within the small subset of selected policies (akin to top-k precision). Or, in scenarios where the practitioner is sensitive to large differences in achieved value, a more relevant downstream metric may be the difference between the largest groundtruth value within the k selected policies compared to the groundtruth of the best possible policy overall (akin to top-k regret). With these potential offline policy selection metrics, it is far from obvious that ranking according to OPE mean or high-confidence bound estimates is ideal [17].

The diversity of downstream metrics for offline policy selection presents a challenge to any algorithm that produces a point estimate for each policy. In fact, any one approach to computing point estimates will necessarily be sub-optimal for some adversarially chosen policy selection criteria. To circumvent this challenge, we propose to compute a *belief distribution* over groundtruth values for each policy. Specifically, with the posteriors of the policy values, one can calculate the distribution of a variety of criteria over the value for each policy. These posteriors can be used in a straightforward procedure that takes estimation uncertainty into account to rank the policy candidates. While this belief distribution approach to offline policy selection is attractive, it also presents its own challenge: how should one estimate such a distribution in the purely offline setting?

We propose *Bayesian Distribution Correction Estimation (BayesDICE)* to address this challenge. BayesDICE works by estimating posteriors over correction ratios for each state-action pair, corresponding to a belief distribution over density ratios between the off-policy data and the stationary distribution of the target policy. In contrast to the point estimates of state-of-the-art DICE estimators [45, 69, 66], BayesDICE maintains a distribution from which the sampled ratio satisfies the stationary distribution condition *with high probability*. Given belief distributions over these correction ratios, the belief distribution over a policy value may be estimated by averaging these correction distributions over offline data, weighted by rewards or other nonlinear utilities in the case of more exotic downstream policy selection criteria.

As a preliminary experiment, we show that the proposed BayesDICE is highly competitive to existing frequentist approaches when applied to confidence interval estimation. Then, we demonstrate the superiority of BayesDICE applied to offline policy selection under different utility measures, across a variety of discrete and continuous RL tasks. Our policy selection experiments suggest that, while conventional wisdom in the OPE literature focuses on using lower bound estimates to select policies (due to safety concerns) [36], policy ranking based on the lower bound estimates may not always lead to lower downstream regret. Furthermore, when other metrics of policy selection are considered, such as top-k precision, being able to sample from the posterior enables significantly better policy selection than only having access to the mean or confidence bounds of the estimated policy values.

We note that the offline policy selection problem is distinct from offline policy *optimization* (OPO) [39, 24, 35, 6], where one seeks a policy from a parameterized class that optimizes a pointwise objective without consideration of its performance relative to an ensemble of reference policies. (This distinction will become clear in Section 2 below.) In summary, the contributions of this paper are three-fold:

- We formally define offline policy selection and compare and contrast it to traditional OPE (and OPO).
- We propose BayesDICE for characterizing the posterior of the stationary state-action ratio, derived from the perspective of stochastic constraints.
- We design a simulation-based policy ranking algorithm, *OfflineSelect*, that converts the estimated posteriors from BayesDICE to a ranking of policies with respect to a selection criterion.

2 Offline Policy Selection

We consider an infinite-horizon Markov decision process (MDP) [54] denoted as $\mathcal{M} = \langle S, A, R, T, \mu_0, \gamma \rangle$, which consists of a state space, an action space, a deterministic reward function, a transition probability function, an initial state distribution, and a discount factor $\gamma \in (0, 1]$. For simplicity, we restrict our analysis to deterministic rewards, and extending our methods to stochastic reward scenarios is straightforward. In this setting, a policy $\pi(a_t|s_t)$ interacts with the environment starting at $s_0 \sim \mu_0$ and receives a scalar reward $r_t = R(s_t, a_t)$ as the environment transitions into a new state $s_{t+1} \sim T(s_t, a_t)$ at each timestep t. The value of a policy is defined as

$$\rho(\pi) := (1-\gamma)\, \mathbb{E}_{s_0, a_t, s_t}\left[\textstyle\sum_{t=0}^{\infty} \gamma^t r_t\right]. \qquad (1)$$

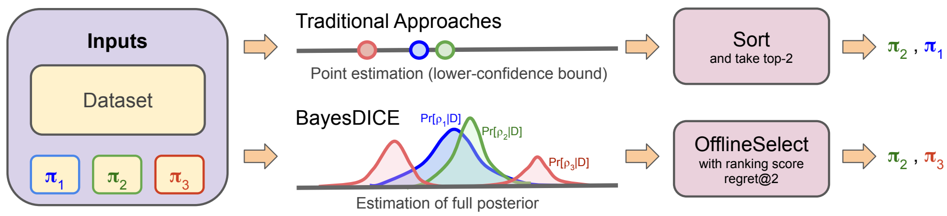

Figure 1: An overview of our proposed approach to offline policy selection. While traditional approaches compute a point estimate for the value of each policy and then rank according to these estimates, BayesDICE approximates an entire belief distribution over the value of each policy conditioned on the provided finite experience dataset. The BayesDICE approximate posteriors are passed to `OfflineSelect` (Algorithm 1), which simulates samples from the posteriors and chooses the policy ranking which achieves the best expected utility (top-2 regret in this example). In many scenarios, leveraging the belief distribution leads to better policy selection than traditional approaches.

We formalize the *offline policy selection* problem as providing a ranking $\mathcal{O} \in \mathrm{Perm}([1, N])$ over a set of *candidate* policies $\{\pi_i\}_{i=1}^N$ given only a *fixed* dataset $\mathcal{D} = \{x^{(j)} := (s_0^{(j)}, s^{(j)}, a^{(j)}, r^{(j)}, s'^{(j)})\}_{j=1}^n$ where $s_0^{(j)} \sim \mu_0$, $(s^{(j)}, a^{(j)}) \sim d^{\mathcal{D}}$ are samples of an unknown distribution $d^{\mathcal{D}}$, $r^{(j)} = R(s^{(j)}, a^{(j)})$, and $s'^{(j)} \sim T(s^{(j)}, a^{(j)})$. [1]

The vanilla approach to the offline policy selection problem is to characterize the *value* of each policy under some *utility* function $u(\pi)$ and then sort the policies accordingly; *i.e.*,

$$\mathcal{O} \leftarrow \mathrm{ArgSortDescending}(\{u(\pi_i)\}_{i=1}^N).$$

The utility $u(\pi_i)$ is typically the result of an OPE algorithm applied to $\mathcal{D}$ and π_i; *i.e.*, $u(\pi_i)$ is either a mean or lower-confidence bound estimate of the policy's normalized per-step reward in (1).

2.1 Selection evaluation

A proposed ranking $\mathcal{O}$ will eventually be evaluated according to how well its policy ordering aligns with the groundtruth policy values. In this section, we elaborate on several potential forms of this evaluation score.

The groundtruth policy value for π_i is given by $\rho(\pi_i)$, and we use $\overline{\rho}_i$ as shorthand for this expression. As part of the offline policy selection problem, we are given a *ranking score* $\mathcal{S}$, which serves as the downstream selection criterion we want to optimize. The ranking score is a function that produces a scalar evaluation metric given a proposed ranking $\mathcal{O}$ and groundtruth policy values of $\{\overline{\rho}_i\}_{i=1}^N$. The $\mathcal{S}$ can take on many forms and is application specific; *e.g.*,

- **top-k precision**: This is an ordinal ranking score. The score considers the top k policies in terms of groundtruth means $\overline{\rho}_i$ and returns the proportion of these which appear in the top k spots of $\mathcal{O}$.
- **top-k accuracy**: Another ordinal ranking score, this score considers the top-k policies in sorted order in terms of groundtruth means $\overline{\rho}_i$ and returns the proportion of these which appear in the same ordinal location in $\mathcal{O}$.
- **top-k correlation**: Another ordinal ranking score, this represents the Pearson correlation coefficient between the ranking of top-k policies in sorted order in terms of groundtruth means $\rho(\pi_i)$ and the truly best top-k policies.
- **top-k regret**: This is a cardinal ranking score. This score respresents the difference in groundtruth means $\overline{\rho}_i$ between the overall best policy – *i.e.*, $\max_i \overline{\rho}_i$ – and the best policy among the top-k ranked policies – *i.e.*, $\max_{i\in[1,k]} \overline{\rho}_{\mathcal{O}[k]}$.
- **Beyond expected return**: One may define the above ranking scores in terms of statistics of the policy value other than the groundtruth means $\{\overline{\rho}_i\}_{i=1}^N$. For example, in safety-critical applications, one may be concerned with the variance of the policy return. Accordingly, one may define CVaR analogues to top-k precision and regret. For simplicity, we will restrict the discussion in this paper to ranking scores which only depend on the groundtruth expected returns $\{\overline{\rho}_i\}_{i=1}^N$.

[1] This tuple-based representation of the dataset is for notational and theoretical convenience, following [11, 34]. In practice, the dataset is usually presented as finite-length trajectories $\{(s_0^{(j)}, a_0^{(j)}, r_0^{(j)}, s_1^{(j)}, \dots)\}_{j=1}^m$, and this can be processed into a dataset of finite samples from μ_0 and from $d^{\mathcal{D}} \times R \times T$. We further assume, for mathematical simplicity, that the dataset is sampled i.i.d., as is common in the OPE literature [62]. In some cases this may be relaxed by assuming a fast mixing time [45].

2.2 Bayes ranking simulation from the posterior

It is not clear whether ranking according to vanilla OPE (either mean or confidence based) is ideal for any of the ranking scores above, including, for example, top-1 regret in the presence of uncertainty. However, if one has access to an approximate belief distribution over the policy values, one can simply simulate the Bayes risk over all candidate ranks to find a near-optimal ranking [18] with respect to an arbitrary specified downstream ranking score, and we elaborate on this Bayes decision procedure here.

In the ideal case if we have access to the true groundtruth policy values $\{\overline{\rho}_i\}_{i=1}^N$, and the ranking score function $\mathcal{S}$, we can calculate the score value of *any* ranking $\mathcal{O}$ and find the ranking $\mathcal{O}^*$ that optimizes this score. However, we are limited to a finite offline dataset and the full return distributions are unknown. In this offline setting, we propose to instead compute a belief distribution $q(\{\overline{\rho}_i\}_{i=1}^N)$, and then we can optimize over the expected ranking score, *i.e.*,

$$\tilde{\mathcal{O}}^* := \operatorname*{argmin}_{\mathcal{O}} \mathbb{E}_q \left[\mathcal{S}(\mathcal{O}, \{\overline{\rho}_i\}_{i=1}^N) \right] \quad (2)$$

as shown in Algorithm 1. This algorithm computes the Bayes risk by simulating realizations of the groundtruth values $\{\overline{\rho}_i\}_{i=1}^N$ with samples from the belief distribution $q(\{\overline{\rho}_i\}_{i=1}^N)$, and in this way estimates the expected realized ranking score $\mathcal{S}$ over all possible rankings $\mathcal{O}$. As we will show empirically, matching the Bayes selection process (the $\mathcal{S}$ used in Algorithm 1) to the downstream ranking score naturally leads to improved performance. The question left now becomes how to effectively learn a belief distribution over $\{\overline{\rho}_i\}_{i=1}^N$, and this is answered by the BayesDICE algorithm.

Algorithm 1 `OfflineSelect`

Inputs Posteriors $q(\{\overline{\rho}_i\}_{i=1}^N)$, ranking score $\hat{\mathcal{S}}$
Initialize $\mathcal{O}^*$; L^* ▷ `Track best score`
for $\mathcal{O}$ in `Perm`$([1, ..., N])$ **do**
  $L = 0$
  for $j = 1$ to n **do**
    sample $\{\hat{\rho}_i^{(j)}\}_{i=1}^N \sim q(\{\overline{\rho}_i\}_{i=1}^N)$
    ▷ `Sum up sample scores`
    $L = L + \hat{\mathcal{S}}(\{\hat{\rho}_i^{(j)}\}_{i=1}^N, \mathcal{O})$
  end for
  if $L < L^*$ **then**
    ▷ `Update best ranking/score`
    $L^* = L$; $\mathcal{O}^* = \mathcal{O}$
  end if
end for; **return** $\mathcal{O}^*, L^*$

3 BayesDICE

We propose BayesDICE for estimating the belief distribution over $\{\overline{\rho}_i\}_{i=1}^N$. We first investigate alternative characterizations of policy value to justify a representation in terms of stationary density correction ratios (generally known as DICE or marginalized importance weights). These correction ratios are characterized by a set of constraints, one for each state-action pair, which presents a challenge for posterior inference. However, by re-expressing Bayesian inference as an optimization, we bypass this difficulty via *stochastic constraints*, a derivation that is of independent interest. We then apply the resulting *constrained posterior inference* to DICE, yielding a novel estimator that is computationally attractive while supporting a broad range of ranking scores for downstream tasks.

3.1 Alternative Representations of Policy Value

To accomplish offline policy selection one must choose a specific expression to represent the value of a policy. There are several principal requirements for such a representation:

- **Offline**: Since we focus on ranking policies given only *offline* data, the policy value should not depend on on-policy samples or access to a known behavior policy.
- **Versatility**: Since the downstream task may utilize different ranking scores, the policy value representation should be compatible with efficient evaluation of these scores.

With these considerations in mind, we review choices for representing the value of a policy π. Define

$$Q^\pi(s,a) = \mathbb{E}\left[\textstyle\sum_{t=0}^\infty \gamma^t R(s_t,a_t) | s_0 = s, a_0 = a\right] \quad \text{and} \quad d^\pi(s,a) = (1-\gamma)\textstyle\sum_{t=0}^\infty \gamma^t d_t^\pi(s,a),$$
$$\text{with} \quad d_t^\pi(s,a) = \mathbf{P}(s_t = s, a_t = a | s_0 \sim \mu_0, \forall i < t, a_i \sim \pi(\cdot|s_i), s_{i+1} \sim T(\cdot|s_i,a_i)),$$

which are the state-action *value function* and *stationary visitations* of π. These quantities satisfy the recursions

$$Q^\pi(s,a) = R(s,a) + \gamma \cdot \mathcal{P}^\pi Q^\pi(s,a), \quad \text{where} \quad \mathcal{P}^\pi Q(s,a) := \mathbb{E}_{s' \sim T(s,a), a' \sim \pi(s')}[Q(s',a')]; \quad (3)$$
$$d^\pi(s,a) = (1-\gamma)\mu_0(s)\pi(a|s) + \gamma \cdot \mathcal{P}_*^\pi d^\pi(s,a), \quad \text{where} \quad \mathcal{P}_*^\pi d(s,a) := \pi(a|s)\textstyle\sum_{\tilde{s},\tilde{a}} T(s|\tilde{s},\tilde{a}) d(\tilde{s},\tilde{a}). \quad (4)$$

From these identities, the policy value can be expressed in two equivalent ways:

$$\rho(\pi) = (1-\gamma) \cdot \mathbb{E}_{\substack{a_0 \sim \pi(s_0) \\ s_0 \sim \mu_0}} [Q^\pi(s_0, a_0)] \tag{5}$$

$$= \mathbb{E}_{(s,a)\sim d^\pi}[r(s,a)]. \tag{6}$$

Current OPE methods are generally based on one of the representations (1), (5) or (6). For example, importance sampling (IS) estimators [53, 44, 19] are based on (1); LSTDQ [37] is a representative algorithm for fitting Q^π and thus based on (5); the DICE algorithms [66] estimate the stationary density ratio $\zeta^\pi(s,a) := \frac{d^\pi(s,a)}{d^{\mathcal{D}}}$ so that $\rho(\pi) = \mathbb{E}_{d^{\mathcal{D}}}[\zeta^\pi \cdot r]$, and are thus based on (6). To reduce notational clutter, we omit the superscripted π on ζ when it is clear from context.

Among the three representations, the stationary density ratio representation fully supports the stated requirements, and hence is the most promising for the ultimate selection task. First, IS estimators suffer from an exponential growth in variance [42] and require knowledge of the behavior policy. By contrast, the functions Q^π and d^π share common minimax properties [62] and can be estimated without knowledge of the behavior policy enabling *behavior-agnostic* learning. However, Q^π exhibits a linear dependence on $R(s,a)$, hence, even if Q^π is estimated accurately, it is still infeasible to evaluate ranking scores that involve $(1-\gamma)\mathbb{E}\left[\sum_{t=0}^\infty \gamma^t \sigma(r_t)\right]$ with a nonlinear σ (unless one learns a different Q function for each possible ranking score, which may be computationally expensive). By contrast, the stationary density ratios $\zeta(s,a)$ are *independent* of reward, which enables efficient ranking on a variety of downstream ranking scores. For example, in the case of a nonlinear utility σ, the policy value may be easily computed from the stationary density ratio as $\mathbb{E}_{d^{\mathcal{D}}}[\zeta \cdot \sigma(r)]$. Based on these considerations, representing policy value via stationary density ratios best satisfies the requirements: it enjoys statistical advantages for offline setting [67, 29] and is flexible for downstream ranking score calculation. Therefore, we focus on developing a Bayesian estimator for ζ^π.

3.2 Stationary Ratio Posterior Estimation

Typically, a posterior $q(\zeta^\pi|\mathcal{D})$ is defined in terms of a prior $p(\zeta^\pi)$ and likelihood function $p(\mathcal{D}|\zeta^\pi)$ via Bayes' rule *i.e.*, $q(\zeta^\pi|\mathcal{D}) \propto p(\mathcal{D}|\zeta^\pi)\, p(\zeta^\pi)$. However, the posterior can also be equivalently expressed as the result of an optimization problem [63, 68]

$$\min_{q\in\mathcal{P}} \; -\mathbb{E}_{q(\zeta^\pi)}[\log p(\mathcal{D}|\zeta^\pi)] + KL(q\|p), \tag{7}$$

$$= \min_q \; \xi + KL(q\|p), \;\text{ s.t. } q \in \mathcal{P} \cap \left\{\xi = -\mathbb{E}_{q(\zeta^\pi)}[\log p(\mathcal{D}|\zeta^\pi)]\right\}. \tag{8}$$

where $\mathcal{P}$ is the space of valid densities. This optimization interpretation of Bayesian inference has been generalized in well known work on *posterior regularization* and *reguarlized Bayes* [43, 41, 70], which considers more complex loss functions on ξ and richer constraints on the "posterior"

$$\min_{q\in\mathcal{P}(\mathcal{D},\xi)} \lambda U(\xi) + KL(q\|p), \tag{9}$$

where $\mathcal{P}(\mathcal{D},\xi) := \mathcal{P} \cap \Omega(\mathcal{D},\xi)$ with $\Omega(\mathcal{D},\xi)$ as a set defined by data-dependent constraints with slack variable ξ and $U(\cdot)$ a loss function. Although (9) can easily express (8), they key advantage is that the more general formulation allows Bayesian inference to be practically applied in scenarios when the likelihood does not have an explicit, tractable form, or when there are additional constraints that cannot be conveniently encoded in the prior or likelihood [43, 41, 70].

This framing allows us to naturally incorporate constraints arising from the stationary density ratio representation (4). However, previous work only considers *finitely* many constraints on *posterior expectations*, while the constraints for ζ induced by (4) consider each ratio function *individually* on arbitrary $(s,a) \in S \times A$, which can potentially be infinitely many. Therefore, to apply the generalized Bayesian framework (9) to our scenario, we first need to extend the formulation by considering a function space embedding to reduce the number of constraints to finitely many [12, 38, 11], then reformulate these as chance constraints to ensure ζ satisfies the constraints with high probability [47].

Constraints Embedding First, we use a function space embedding to reduce the number of constraints to finitely many [12, 38, 11]. Let $\Delta_d(s,a) := (1-\gamma)\mu_0(s)\pi(a|s) + \gamma\cdot\mathcal{P}_*^\pi d(s,a) - d(s,a)$. Consider a feature mapping $\phi(\cdot,\cdot): S\times A \to \mathbb{R}^m$ and the induced RKHS $\mathcal{H}_\phi$, and define $\langle\phi, \Delta_d\rangle := \mathbb{E}_{(1-\gamma)\mu_0(s)\pi(a|s)+\gamma\cdot\mathcal{P}_*^\pi d(s,a)}[\phi(s,a)] - \mathbb{E}_{d(s,a)}[\phi(s,a)]$.

Then the constraints (4) can be expressed as $\Delta_d(s,a) = 0$. We can match distributions in terms of their embeddings [57] by measuring $\langle\phi,\Delta_d\rangle^\top \langle\phi,\Delta_d\rangle$, a generalization of the approximation methods in [12, 38]. In particular, when $|S|\,|A|$ is finite and we set $\phi(s,a)=\delta_{s,a}$, where $\delta_{s,a}\in\{0,1\}^{|S||A|}$ is an indicator vector with a single 1 at position (s,a) and 0 otherwise, we are matching the distributions pointwise. The feature map $\phi(s,a)$ can also be set to general reproducing kernel $k((s,a),\cdot)\in\mathbb{R}^\infty$. As long as the kernel $k(\cdot,\cdot)$ is characteristic, the embeddings will match if and only if the distributions are identical almost surely [58]. We further re-frame the constraint with Fenchel duality [46]

$$\begin{aligned}\langle\phi,\Delta_d\rangle^\top\langle\phi,\Delta_d\rangle &= \max_{\beta\in\mathcal{H}_\phi}\beta^\top\langle\phi,\Delta_d\rangle-\beta^\top\beta\\ &= \ell(\zeta,\mathcal{D}):=\max_{\beta\in\mathcal{H}_\phi}(1-\gamma)\,\mathbb{E}_{\mu_0\pi}\left[\beta^\top\phi\right]-\beta^\top\beta+\mathbb{E}_{d^{\mathcal{D}}}\left[\zeta(s,a)\,\beta^\top(\gamma\phi(s',a')-\phi(s,a))\right],\end{aligned}\tag{10}$$

resulting in the final constraint $\ell(\zeta,\mathcal{D})=0$.

Chance Constraints Given that the experience is a finite sample from $d^{\mathcal{D}}$, we have to approximate ℓ with a sample estimator $\hat{\ell}$ and the constraint for ζ in (10) might not hold exactly using $\hat{\ell}$. However, under mild conditions, we have $\frac{d^\pi(s,a)}{d^{\mathcal{D}}}\in\Xi:=\left\{\zeta:\hat{\ell}(\zeta,\mathcal{D})\leqslant\epsilon\right\}$ with high probability (see Appendix A for the precise statement and proof). Thus, we expect a randomly sampled ratio $\zeta\sim q(\zeta)$ to be in the relaxed feasible set Ξ with high probability. Incorporating this into (9) yields

$$\min_q\ KL(q\|p)-\lambda\xi,\quad \text{s.t. }\ \mathbb{P}_q(\ell(\zeta,\mathcal{D})\leqslant\epsilon)\geqslant\xi,\tag{11}$$

where the chance constraint enforces the probability that ζ is feasible under the posterior. This formulation can be equivalently rewritten as

$$\min_q\qquad KL(q\|p)-\lambda\mathbb{P}_q(\ell(\zeta,\mathcal{D})\leqslant\epsilon)\tag{12}$$

Then, by applying Markov's inequality, *i.e.*, $\mathbb{P}_q(\ell(\zeta,\mathcal{D})\leqslant\epsilon)=1-\mathbb{P}_q(\ell(\zeta,\mathcal{D})\geqslant\epsilon)\geqslant 1-\frac{\mathbb{E}_q[\ell(\zeta,\mathcal{D})]}{\epsilon}$, we can obtain an upper bound on (12) as

$$\min_q\ KL(q\|p)+\frac{\lambda}{\epsilon}\mathbb{E}_q[\ell(\zeta,\mathcal{D})]\tag{13}$$

$$\begin{aligned}=\min_{q(\zeta)}\max_{q(\beta|\zeta)}KL(q\|p)+\frac{\lambda}{\epsilon}\mathbb{E}_{q(\zeta)q(\beta|\zeta)}\Big[\hat{\mathbb{E}}_{\mathcal{D}}\left[\zeta(s,a)\cdot\beta^\top\left(\gamma\phi(s',a')-\phi(s,a)\right)-f^*(\beta)\right]\\ +(1-\gamma)\,\mathbb{E}_{\mu_0\pi}\left[\beta^\top\phi\right]\Big],\end{aligned}\tag{14}$$

where the last equation follows by interchangeability [56, 10]. Note that $\ell(\zeta,\mathcal{D})\geqslant 0$ since $\mathcal{H}_\phi$ is symmetric, so the outer optimization is lower bounded. We amortize the optimization for β w.r.t. each ζ to a distribution $q(\beta|\zeta)$ to reduce the computational effort. The pseudo-code of the BayesDICE algorithm is shown in Algorithm 2.

Finally, with the posterior approximation for ζ_i, denoting the estimate for candidate policy i, we can draw posterior samples of $\bar{\rho}_i$ by drawing a sample $\zeta_i\sim q(\zeta_i)$ and computing $\hat{\rho}_i=\frac{1}{n}\sum_{(s,a,r)\in\mathcal{D}}\zeta_i(s,a)r$. This defines a posterior distribution over $\bar{\rho}_i$. For the joint posterior over $\{\bar{\rho}_i\}_{i=1}^N$ we use a mean field approximation to express it as a product of independent marginals, *i.e.*, $q(\{\bar{\rho}_i\}_{i=1}^N)=\prod_i q(\bar{\rho}_i)$. This defines the necessary inputs for `OfflineSelect` to determine a ranking of the candidate policies.

Given the space limits, please see Appendix B and C for a discussion of other important aspects of BayesDICE, including an alternative safe surrogate of the chance constraints, parametrization of the posteriors, variants of BayesDICE for undiscounted MDPs, connections to vanilla Bayesian stochastic processes, and the application of BayesDICE to exploration.

4 Related work

We categorize the relevant related work into five categories: offline policy selection, offline policy optimization, off-policy evaluation, Bayesian reinforcement learning, and posterior regularization.

Offline policy selection The decision making problem we formalize as offline policy selection is a member of a set of problems in RL referred to as *model selection*. Previously, this term has been used to refer to state abstraction selection [28, 30] as well as learning algorithm and feature selection [23, 50]. More relevant to our proposed notion of policy selection are a number of previous

Algorithm 2 BayesDICE

Inputs sampled initial states $\hat{\mu}_0 = \{s_0^{(j)}\}_{j=1}^m$, offline data $\mathcal{D} = \{(s_0^{(j)}, s^{(j)}, a^{(j)}, r^{(j)}, s'^{(j)})\}_{j=1}^n$, target policy π, parametrized distributions $q_{\theta_1}(\cdot,\cdot)$ and $q_{\theta_2}(\cdot,\cdot)$, a prior p, convex function f (conjugate f^*), constants ϵ, λ, learning rates η_ζ, η_β, training iterations T, and batch size B.
for $t = 1, \dots, T$ **do**
  Sample batch $\{(s^{(j)}, a^{(j)}, r^{(j)}, s'^{(j)})\}_{j=1}^B$ from $\mathcal{D}$, $\{s_0^{(j)}\}_{j=1}^B$ from $\hat{\mu}_0$, $a'^{(j)} \sim \pi(s'^{(j)})$ and $a_0^{(j)} \sim \pi(s_0^{(j)})$ for $j = 1, \dots, B$.
  Sample $\beta_0 \sim q_{\theta_1}(s_0^{(j)}, a_0^{(j)})$, $\beta \sim q_{\theta_1}(s^{(j)}, a^{(j)})$, $\beta' \sim q_{\theta_1}(s'^{(j)}, a'^{(j)})$, and $\zeta \sim q_{\theta_2}(s^{(j)}, a^{(j)})$.
  Compute loss $\hat{J} = KL(p\|q_{\theta_1}) + KL(p\|q_{\theta_2}) + \frac{\lambda}{\epsilon B}\sum_{i=1}^B(\zeta\gamma(\beta - \beta') - f^*(\beta)) + (1-\gamma)\beta_0$.
  Update $\theta_1 \leftarrow \theta_1 + \eta_\beta \nabla_{\theta_1}\hat{J}$ and $\theta_2 \leftarrow \theta_2 - \eta_\zeta \nabla_{\theta_2}\hat{J}$.
end for; **return** $q_{\theta_2}(\cdot,\cdot)$

works which use model selection to refer to the problem of choosing a near-optimal Q-function from a set of candidate approximation functions [21, 20, 27, 64]. In this case, the evaluation metric is typically defined as the L_∞ norm of difference of Q versus the state-action value function of the optimal policy Q^*. While one can relate this evaluation metric to the sub-optimality (*i.e.*, regret) of the policy induced by the Q-function, we argue that our proposed policy selection problem is both more general – since we allow for the use of policy evaluation metrics other than sub-optimality – and more practically relevant – since in many practical applications, the policy may not be expressible as the argmax of a Q-function. Lastly, the offline policy selection problem we describe is arguably a formalization of the problem approached in [51] and referred to as *hyperparameter selection*. In contrast to this previous work, we not only formalize the decision problem, but also propose a method to directly optimize the policy selection evaluation metric. Offline policy selection has also been studied by [17], who consider desirable properties of a point estimator to yield good rankings in terms of a notion of ranking score referred to as *fairness*.

Offline policy optimization While it is possible to integrate desired criteria such as pessimism into offline policy optimization [35, 6], this requires the desired criteria (e.g., maximum high-confidence lower bound) to be specified prior to policy learning, which might differ from what a practitioner deploying the policy prefers (e.g., policies that achieve top-k precision or regret). Furthermore, policies in practical applications may not be amenable to (policy)-gradient-based learning (e.g., policies with business logic and hard-coded rules). In these cases, it is much easier to rank a set of candidate policies given a set of criteria rather than learning one policy for each criterion.

Off-policy evaluation Off-policy evaluation (OPE) is a highly active area of research. While the original motivation for OPE was in the pursuit of policy selection [53, 28], the field has historically almost exclusively focused on the related but distinct problem of estimating the online value (accumulated rewards) of a single target policy. In addition to a plethora of techniques for providing point estimates of this groundtruth value [19, 4, 31, 59, 32, 45, 69, 66], there is also a growing body of literature that uses frequentist principles to derive high-confidence lower bounds for the value of a policy [4, 61, 25, 36, 22, 11, 34]. As our results demonstrate, ranking or selecting policies based on either their estimated mean or lower confidence bounds can at times be sub-optimal, depending on the evaluation criteria.

Bayesian reinforcement learning Our proposed method for offline policy selection relies on Bayesian principles to estimate a posterior distribution over the groundtruth policy value. While many Bayesian RL methods have been proposed for policy optimization [14, 52], especially in the context of exploration [26, 13, 33], relatively few have been proposed for policy evaluation. In one instance, [21] derive PAC-Bayesian bounds on estimates of the Bellman error of a candidate Q-value function. In contrast to this work, the BayesDICE estimates a distribution over stationary density ratio, and this distribution allows us to directly optimize arbitrary downstream policy selection metrics.

Distinguish distributional RL Although both distributional RL [2, 8, 7] and BayesDICE learn distributions over quantities of interest, these distributions are significantly different and with different update rules. Distributional RL fits a distribution of returns over future trajectories, where the randomness comes from stochasticity of MDP transitions and policy action selections. In contrast,

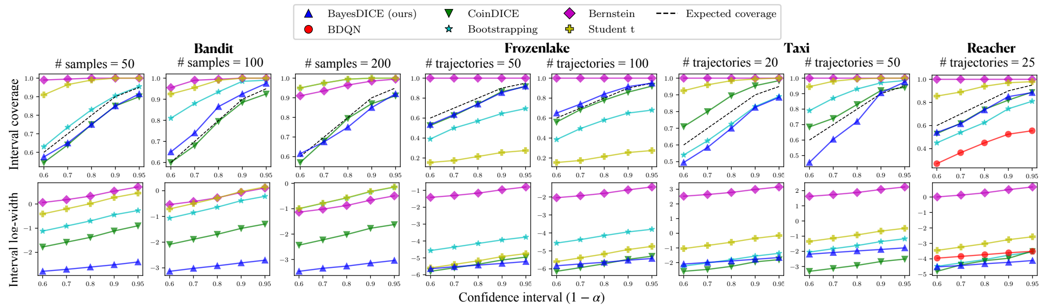

Figure 2: CI estimation results. The y-axis shows the empirical coverage and median log-interval width across 200 trials. BayesDICE exhibits near true coverage with narrow interval width.

BayesDICE learns distributions of stationary density ratios in a *Bayesian posterior* sense, which captures uncertainty from both model stochasticity and finite observations, while marginalizing over any stochasticity in MDP transitions and policy action selections. More importantly, BayesDICE is designed to serve as a component for policy selection derived via Bayes decision theory, with which distributional RL is not compatible.

Bayesian inference with posterior regularization Unlike vanilla Bayesian inference for posterior computation, the proposed BayesDICE does not rely on an explicitly computed log-likelihood, but instead estimates the posterior of the stationary density ratio by enforcing a stochastic constraint. This formulation of BayesDICE is inspired by the functional optimization view of Bayesian inference [63, 68, 9]. There are several works introducing the data-dependent constraints or regularization to encode the side information of the posterior into the optimization framework, *e.g.*, generalized expectation criteria [43], learning from measurements [41], and regularized Bayes [70]. The most important difference lies in the formulation of the constraints: the existing works only considers *expectation constraints/regularization*, while we largely extend the framework to more general *chance constraints*.

5 Experiments

We empirically evaluate BayesDICE in estimating confidence intervals (which can be used for policy selection) and offline policy selection under linear and neural network posterior parametrizations on tabular Bandit, Taxi [16], FrozenLake [5], and continuous-control Reacher [5] tasks. As shown in Figure 2, BayesDICE outperforms existing methods for confidence interval (CI) estimation based on concentration inequalities, producing accurate coverage while maintaining tight interval width, suggesting that BayesDICE achieves accurate posterior estimation in practice while being robust to approximation errors and potentially misaligned Bayesian priors. Moreover, in offline policy selection settings, matching the selection criteria (Algorithm 1) to a variety of ranking scores (enabled by the estimated posterior) shows clear advantage over policy ranking based on point estimates or confidence intervals. See Appendix D for additional results and implementation details.

5.1 Confidence interval estimation

We first evaluate the BayesDICE approximate posterior by computing the accuracy of the *credible* intervals [40] it produces. To make comparisons with previous work, we evaluate frequentist confidence interval properties of BayesDICE against a known set of CI estimators based on concentration inequalities, and against CoinDICE [11], which is based-on empirical likelihood. While the frequentist confidence interval is analagous to the Bayesian *credible interval*, they have different statistical properties, so we expect that evaluating the credible intervals BayesDICE produces under frequentist measures will give a pessimistic estimate of its true performance. To compute the concentration-inequality-based baselines, we follow [11] by first using weighted (*i.e.*, self-normalized) per-step importance sampling [59] to obtain a policy value estimate for each logged trajectory. These trajectories provide a finite sample of value estimates. We use self-normalized importance sampling in MDP environments (which has been found to yield better empirical results on these tasks [42, 45] despite being biased). We then use empirical *Bernstein's* inequality [61], bias-corrected *bootstrap* [60], and *Student's t-test* to derive lower and upper high-confidence bounds on these estimates. We further consider Bayesian Deep Q-Networks (BDQN, only applicable to function approximation) [1] with an

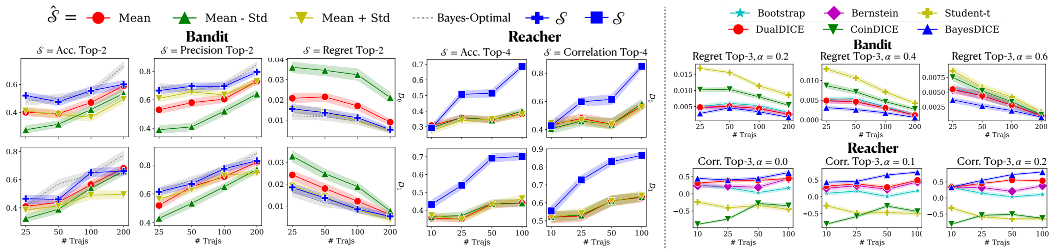

Figure 3: Left: Policy selection using top-k ranking scores compared to mean/confidence ranking approaches on two-armed Bandit and Reacher. We fix the posterior to the one approximated by BayesDICE and evaluate different $\hat{\mathcal{S}}$ used in Algorithm 1 to compute a policy ranking. Using $\hat{\mathcal{S}} = \mathcal{S}$ results in the best performace. Right: Policy selection under regret and correlation at top-k compared to other methods using point estimate (DualDICE) or high-confidence lower bounds. Mean and standard error across 10 seeds are shown.

average empirical reward prior in the function approximation setting. BDQN applies Bayesian linear regression to the last layer of a deep Q-network to learn a distribution of Q-values. Both BayesDICE and BDQN output a distribution of parameters, from which we conduct Monte Carlo sampling and use the resulting samples to compute a credible interval at a given confidence level.

We plot the empirical coverage and interval width at different confidence levels in Figure 2. To compute the empirical *interval coverage*, we conduct 200 trials with randomly sampled datasets. The interval coverage is the proportion of the 200 intervals that contains the true value of the target policy. The *interval log-width* is the median of the log width of the 200 intervals. As shown in Figure 2, BayesDICE's coverage closely follows the intended coverage (black dotted line), while maintaining narrow interval width across all tasks.

5.2 Policy selection

Next, we demonstrate the benefit of matching the policy selection criteria to the ranking score in offline policy selection. Our evaluation is based on a variety of cardinal and ordinal ranking scores defined in Section 2.1. We begin by considering the use of Algorithm 1 with BayesDICE-approximated posteriors. By keeping the BayesDICE posterior fixed, we focus our evaluation on the performance of Algorithm 1. We plot the groundtruth performance of this procedure applied to Bandit and Reacher in Figure 3 (left). These figures compare using different $\hat{\mathcal{S}}$ to rank the policies according to Algorithm 1 across different downstream ranking scores $\mathcal{S}$. We find that aligning the criteria $\hat{\mathcal{S}}$ used in Algorithm 1 with the downstream ranking score $\mathcal{S}$ is empirically the best approach ($\hat{\mathcal{S}} = \mathcal{S}$). In contrast, using point estimates such as `Mean` or `Mean` $\pm$ `Std` can yield much worse downstream performance. We also see that in the Bandit setting, where we can analytically compute the Bayes-optimal ranking, using aligned ranking scores in conjunction with BayesDICE-approximated posteriors achieves near-optimal performance.

Having established BayesDICE's ability to compute accurate posterior distributions as well as the benefit of appropriately aligning the ranking score used in Algorithm 1, we compare BayesDICE to state-of-the-art OPE methods in policy selection. In these experiments, we use Algorithm 1 with posteriors approximated by BayesDICE and $\hat{\mathcal{S}} = \mathcal{S}$. We compare the use of BayesDICE in this way to ranking via point estimates of DualDICE [45] and other confidence-interval estimation methods introduced in Section 5.1. We present results in Figure 3, in terms of top-k regret and correlation on Bandit and Reacher tasks across different sample sizes and behavior data. BayesDICE outperforms other methods on both tasks. See additional ranking results in Appendix D.

6 Conclusion

In this paper, we formally defined the offline policy selection problem, and proposed BayesDICE to first estimate posterior distributions of policy values before using a simulation-based procedure to compute an optimal policy ranking. Empirically, BayesDICE not only provides accurate belief distribution estimation, but also shows excellent performance in policy selection tasks. Extending BayesDICE to estimating a posterior distribution over return distributions (instead of the expected return) is an important direction of future research.

References

[1] Kamyar Azizzadenesheli, Emma Brunskill, and Animashree Anandkumar. Efficient exploration through Bayesian deep Q-networks. In *2018 Information Theory and Applications Workshop (ITA)*, pages 1–9. IEEE, 2018.

[2] Marc G Bellemare, Will Dabney, and Rémi Munos. A distributional perspective on reinforcement learning. *arXiv preprint arXiv:1707.06887*, 2017.

[3] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust optimization*. Princeton University Press, 2009.

[4] Léon Bottou, Jonas Peters, Joaquin Quiñonero-Candela, Denis X. Charles, D. Max Chickering, Elon Portugaly, Dipankar Ray, Patrice Simard, and Ed Snelson. Counterfactual reasoning and learning systems: The example of computational advertising. *Journal of Machine Learning Research*, 14(65):3207–3260, 2013.

[5] Greg Brockman, Vicki Cheung, Ludwig Pettersson, Jonas Schneider, John Schulman, Jie Tang, and Wojciech Zaremba. OpenAI Gym. *arXiv preprint arXiv:1606.01540*, 2016.

[6] Jacob Buckman, Carles Gelada, and Marc Bellemare. The importance of pessimism in fixed-dataset policy optimization. In *International Conference on Learning Representations (ICLR)*, 2021.

[7] Will Dabney, Georg Ostrovski, David Silver, and Rémi Munos. Implicit quantile networks for distributional reinforcement learning. *arXiv preprint arXiv:1806.06923*, 2018.

[8] 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.

[9] Bo Dai, Niao He, Hanjun Dai, and Le Song. Provable Bayesian inference via particle mirror descent. In *Proceedings of the 19th International Conference on Artificial Intelligence and Statistics*, pages 985–994, 2016.

[10] Bo Dai, Niao He, Yunpeng Pan, Byron Boots, and Le Song. Learning from conditional distributions via dual embeddings. In *Artificial Intelligence and Statistics*, pages 1458–1467, 2017.

[11] Bo Dai, Ofir Nachum, Yinlam Chow, Lihong Li, Csaba Szepesvári, and Dale Schuurmans. Coindice: Off-policy confidence interval estimation. *arXiv preprint arXiv:2010.11652*, 2020.

[12] Daniela Pucci De Farias and Benjamin Van Roy. The linear programming approach to approximate dynamic programming. *Operations research*, 51(6):850–865, 2003.

[13] Richard Dearden, Nir Friedman, and David Andre. Model-based Bayesian exploration. *arXiv preprint arXiv:1301.6690*, 2013.

[14] Marc Deisenroth and Carl E Rasmussen. Pilco: A model-based and data-efficient approach to policy search. In *Proceedings of the 28th International Conference on machine learning (ICML-11)*, pages 465–472, 2011.

[15] David Dereudre. Introduction to the theory of Gibbs point processes. In *Stochastic Geometry*, pages 181–229. Springer, 2019.

[16] Thomas G Dietterich. The MAXQ method for hierarchical reinforcement learning. In *ICML*, volume 98, pages 118–126. Citeseer, 1998.

[17] Shayan Doroudi, Philip S Thomas, and Emma Brunskill. Importance sampling for fair policy selection. *Grantee Submission*, 2017.

[18] Richard O. Duda, Peter E. Hart, and David G. Stork. *Pattern Classification (2nd Edition)*. Wiley-Interscience, 2 edition, November 2000.

[19] Miroslav Dudík, John Langford, and Lihong Li. Doubly robust policy evaluation and learning. *arXiv preprint arXiv:1103.4601*, 2011.

[20] Amir-massoud Farahmand and Csaba Szepesvári. Model selection in reinforcement learning. *Machine learning*, 85(3):299–332, 2011.

[21] Mahdi M Fard and Joelle Pineau. PAC-Bayesian model selection for reinforcement learning. In *Advances in Neural Information Processing Systems*, pages 1624–1632, 2010.

[22] Yihao Feng, Tongzheng Ren, Ziyang Tang, and Qiang Liu. Accountable off-policy evaluation with kernel Bellman statistics. *arXiv preprint arXiv:2008.06668*, 2020.

[23] Dylan J Foster, Akshay Krishnamurthy, and Haipeng Luo. Model selection for contextual bandits. In *Advances in Neural Information Processing Systems*, pages 14741–14752, 2019.

[24] Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *Proceedings of the International Conference on Machine Learning*, 2019.

[25] Josiah P Hanna, Peter Stone, and Scott Niekum. Bootstrapping with models: Confidence intervals for off-policy evaluation. *arXiv preprint arXiv:1606.06126*, 2016.

[26] Rein Houthooft, Xi Chen, Yan Duan, John Schulman, Filip De Turck, and Pieter Abbeel. Vime: Variational information maximizing exploration. In *Advances in Neural Information Processing Systems*, pages 1109–1117, 2016.

[27] Alexander Irpan, Kanishka Rao, Konstantinos Bousmalis, Chris Harris, Julian Ibarz, and Sergey Levine. Off-policy evaluation via off-policy classification. In *Advances in Neural Information Processing Systems*, pages 5437–5448, 2019.

[28] Nan Jiang. *A Theory of Model Selection in Reinforcement Learning*. PhD thesis, 2017.

[29] Nan Jiang and Jiawei Huang. Minimax value interval for off-policy evaluation and policy optimization. *Advances in Neural Information Processing Systems*, 33, 2020.

[30] Nan Jiang, Alex Kulesza, and Satinder Singh. Abstraction selection in model-based reinforcement learning. In *International Conference on Machine Learning*, pages 179–188, 2015.

[31] Nan Jiang and Lihong Li. Doubly robust off-policy value evaluation for reinforcement learning. *arXiv preprint arXiv:1511.03722*, 2015.

[32] Nathan Kallus and Masatoshi Uehara. Double reinforcement learning for efficient off-policy evaluation in Markov decision processes. *Journal of Machine Learning Research*, 21(167):1–63, 2020.

[33] J Zico Kolter and Andrew Y Ng. Near-Bayesian exploration in polynomial time. In *Proceedings of the 26th annual international conference on machine learning*, pages 513–520, 2009.

[34] Ilya Kostrikov and Ofir Nachum. Statistical bootstrapping for uncertainty estimation in off-policy evaluation, 2020.

[35] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative Q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

[36] Ilja Kuzborskij, Claire Vernade, András György, and Csaba Szepesvári. Confident off-policy evaluation and selection through self-normalized importance weighting. *arXiv preprint arXiv:2006.10460*, 2020.

[37] Michail G Lagoudakis and Ronald Parr. Least-squares policy iteration. *Journal of machine learning research*, 4(Dec):1107–1149, 2003.

[38] Chandrashekar Lakshminarayanan, Shalabh Bhatnagar, and Csaba Szepesvári. A linearly relaxed approximate linear program for Markov decision processes. *CoRR*, abs/1704.02544, 2017.

[39] Sascha Lange, Thomas Gabel, and Martin Riedmiller. Batch reinforcement learning. In *Reinforcement Learning*, pages 45–73. 2012.

[40] Peter M. Lee. *Bayesian Statistics: An Introduction*. Wiley, 1997.

[41] Percy Liang, Michael I Jordan, and Dan Klein. Learning from measurements in exponential families. In *Proceedings of the 26th annual international conference on machine learning*, pages 641–648, 2009.

[42] Qiang Liu, Lihong Li, Ziyang Tang, and Dengyong Zhou. Breaking the curse of horizon: Infinite-horizon off-policy estimation. In *Advances in Neural Information Processing Systems*, pages 5356–5366, 2018.

[43] Gideon S Mann and Andrew McCallum. Generalized expectation criteria for semi-supervised learning with weakly labeled data. *Journal of machine learning research*, 11(2), 2010.

[44] S. Murphy, M. van der Laan, and J. Robins. Marginal mean models for dynamic regimes. *Journal of American Statistical Association*, 96(456):1410–1423, 2001.

[45] Ofir Nachum, Yinlam Chow, Bo Dai, and Lihong Li. DualDICE: Behavior-agnostic estimation of discounted stationary distribution corrections. In *Advances in Neural Information Processing Systems*, pages 2315–2325, 2019.

[46] Ofir Nachum and Bo Dai. Reinforcement learning via Fenchel-Rockafellar duality. *arXiv preprint arXiv:2001.01866*, 2020.

[47] Arkadi Nemirovski and Alexander Shapiro. Convex approximations of chance constrained programs. *SIAM Journal on Optimization*, 17(4):969–996, 2007.

[48] Brendan ODonoghue, Ian Osband, Remi Munos, and Volodymyr Mnih. The uncertainty Bellman equation and exploration. In *International Conference on Machine Learning*, pages 3836–3845, 2018.

[49] Ian Osband, Benjamin Van Roy, Daniel J Russo, and Zheng Wen. Deep exploration via randomized value functions. *Journal of Machine Learning Research*, 20(124):1–62, 2019.

[50] Aldo Pacchiano, My Phan, Yasin Abbasi-Yadkori, Anup Rao, Julian Zimmert, Tor Lattimore, and Csaba Szepesvari. Model selection in contextual stochastic bandit problems. *arXiv preprint arXiv:2003.01704*, 2020.

[51] Tom Le Paine, Cosmin Paduraru, Andrea Michi, Caglar Gulcehre, Konrad Zolna, Alexander Novikov, Ziyu Wang, and Nando de Freitas. Hyperparameter selection for offline reinforcement learning. *arXiv preprint arXiv:2007.09055*, 2020.

[52] Paavo Parmas, Carl Edward Rasmussen, Jan Peters, and Kenji Doya. Pipps: Flexible model-based policy search robust to the curse of chaos. In *International Conference on Machine Learning*, pages 4065–4074. PMLR, 2018.

[53] Doina Precup, Richard S. Sutton, and Satinder P. Singh. Eligibility traces for off-policy policy evaluation. In *Proceedings of the 17th International Conference on Machine Learning*, pages 759–766, 2000.

[54] Martin L Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., 1994.

[55] Ali Rahimi and Benjamin Recht. Weighted sums of random kitchen sinks: Replacing minimization with randomization in learning. *Advances in neural information processing systems*, 21:1313–1320, 2008.

[56] Alexander Shapiro, Darinka Dentcheva, and Andrzej Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory*. SIAM, 2014.

[57] Alex Smola, Arthur Gretton, Le Song, and Bernhard Schölkopf. A Hilbert space embedding for distributions. In *International Conference on Algorithmic Learning Theory*, pages 13–31. Springer, 2007.

[58] Bharath K Sriperumbudur, Kenji Fukumizu, and Gert RG Lanckriet. Universality, characteristic kernels and RKHS embedding of measures. *Journal of Machine Learning Research*, 12(7), 2011.

[59] Philip Thomas and Emma Brunskill. Data-efficient off-policy policy evaluation for reinforcement learning. In *International Conference on Machine Learning*, pages 2139–2148, 2016.

[60] Philip Thomas, Georgios Theocharous, and Mohammad Ghavamzadeh. High confidence policy improvement. In *International Conference on Machine Learning*, pages 2380–2388, 2015.

[61] Philip S Thomas, Georgios Theocharous, and Mohammad Ghavamzadeh. High-confidence off-policy evaluation. In *Twenty-Ninth AAAI Conference on Artificial Intelligence*, 2015.

[62] Masatoshi Uehara and Nan Jiang. Minimax weight and Q-function learning for off-policy evaluation. 2020.

[63] Peter M Williams. Bayesian conditionalisation and the principle of minimum information. *The British Journal for the Philosophy of Science*, 31(2):131–144, 1980.

[64] Tengyang Xie and Nan Jiang. Batch value-function approximation with only realizability, 2020.

[65] Mengjiao Yang, Bo Dai, Hanjun Dai, and Dale Schuurmans. Energy-based processes for exchangeable data. *arXiv preprint arXiv:2003.07521*, 2020.

[66] Mengjiao Yang, Ofir Nachum, Bo Dai, Lihong Li, and Dale Schuurmans. Off-policy evaluation via the regularized Lagrangian. In *Advances in Neural Information Processing Systems*, 2020.

[67] Ming Yin and Yu-Xiang Wang. Asymptotically efficient off-policy evaluation for tabular reinforcement learning. In *International Conference on Artificial Intelligence and Statistics*, pages 3948–3958. PMLR, 2020.

[68] Arnold Zellner. Optimal Information Processing and Bayes's Theorem. *The American Statistician*, 42(4), November 1988.

[69] Ruiyi Zhang, Bo Dai, Lihong Li, and Dale Schuurmans. GenDICE: Generalized offline estimation of stationary values. In *International Conference on Learning Representations*, 2020.

[70] Jun Zhu, Ning Chen, and Eric P Xing. Bayesian inference with posterior regularization and applications to infinite latent SVMs. *The Journal of Machine Learning Research*, 15(1):1799–1847, 2014.

Checklist

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

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

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

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] Future direction in Section 6

(c) Did you discuss any potential negative societal impacts of your work? [N/A]

(d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]

2. If you are including theoretical results...

   (a) Did you state the full set of assumptions of all theoretical results? [Yes] See Appendix A

   (b) Did you include complete proofs of all theoretical results? [Yes] See Appendix A

3. If you ran experiments...

   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] Code submitted in supplementary.

   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Appendix D

   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [Yes] See caption of Figure 2 and Figure 3

   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Appendix D

4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...

   (a) If your work uses existing assets, did you cite the creators? [Yes] See Section 5

   (b) Did you mention the license of the assets? [Yes] The license is Apache License 2.0.

   (c) Did you include any new assets either in the supplemental material or as a URL? [No]

   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]

   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]

5. If you used crowdsourcing or conducted research with human subjects...

   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]

   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]

Appendix

A Proofs for Finite Sample Relaxation

The following lemma will be needed.

Lemma 1. *[[55], Lemma 4] Let $\mathbf{X} = \{x_i\}_{i=1}^n$ be* i.i.d.*random variables in a ball $\mathcal{H}$ of radius C centered around the origin in a Hilbert space. Denote their average by $\bar{x} = \frac{1}{n}\sum_{i=1}^n x_i$. Then for any $\delta > 0$, with probability at least $1-\delta$,*

$$\|\bar{x} - \mathbb{E}\bar{x}\| \leqslant \frac{M}{\sqrt{n}}\left(1 + \sqrt{2\log\frac{1}{\delta}}\right).$$

Theorem 2. *Denote $\zeta^*(s,a) = \frac{d^\pi(s,a)}{d^\mathcal{D}}$ which is bounded by C_ζ, under the assumption that $\|\phi\|_2 \leqslant C_\phi$, $\|\beta\|_2 \leqslant C_\beta$, $\forall \beta \in \mathcal{H}_\beta$ and f is L_f-Lipschitz continuous, then $\zeta^* \in \Xi := \{\zeta : \ell(\zeta, \mathcal{D}) \leqslant \epsilon\}$ with probability $1 - \exp\left(-\frac{n\epsilon^2}{2C}\right)$ with $C := (1+\gamma)(1+C_\zeta) C_\beta C_\phi + L_f C_\beta$.*

Proof. Let

$$\iota(\zeta, \mathcal{D}, \beta) := (1-\gamma)\mathbb{E}_{\mu_0\pi}\left[\beta^\top \phi\right] + \hat{\mathbb{E}}_\mathcal{D}\left[\zeta(s,a)\cdot\beta^\top(\gamma\phi(s',a') - \phi(s,a)) - f^*(\beta)\right],$$

and

$$\iota\left(\zeta, d^\mathcal{D}, \beta\right) := (1-\gamma)\mathbb{E}_{\mu_0\pi}\left[\beta^\top \phi\right] + \mathbb{E}_{d^\mathcal{D}}\left[\zeta(s,a)\cdot\beta^\top(\gamma\phi(s',a') - \phi(s,a)) - f^*(\beta)\right].$$

We also denote $\hat{\beta} = \operatorname{argmax}_{\beta\in\mathcal{H}_\phi} \iota(\zeta, \mathcal{D}, \beta)$.

Following the discussion in footnote 2 in main text, the $\mathcal{D} \sim d^\mathcal{D}$ *i.i.d.*, it is obvious that $\mathbb{E}[\iota(\zeta,\mathcal{D},\beta)] = \iota\left(\zeta, d^\mathcal{D}, \beta\right)$. Under the bounded assumption of (β, ϕ), we can bound $\|\iota\|_\infty \leqslant C$. Therefore, by Lemma 1, we have

$$P\left(\iota\left(\zeta^*, \mathcal{D}, \hat{\beta}\right) - \iota\left(\zeta^*, \mathcal{D}, \hat{\beta}\right) \geqslant \epsilon\right) \leqslant \exp\left(-\frac{n\epsilon^2}{2C}\right).$$

Since $\zeta^*(s,a) = \frac{d^\pi(s,a)}{d^\mathcal{D}}$, we have $\iota\left(\zeta^*, d^\mathcal{D}, \beta\right) = 0$, $\forall \beta \in \mathcal{H}_\phi$. Finally, recall $\max_{\beta\in\mathcal{H}_\phi} \iota(\zeta, \mathcal{D}, \beta) \geqslant 0$ since $\mathcal{H}_\phi$ is symmetric. We achieve the conclusion. □

B More Discussions on BayesDICE

In this section, we provide more details about BayesDICE.

Remark (Alternative safe surrogates of chance constraints): We apply the Markov's inequality to (12) for the upper bound (13). In fact, the optimization with chance constraints has rich literature [3], where plenty of surrogates can be derived with different safe approximations. For example, if the parametrization of q is simple, one can directly calculate the CDF for the probability $\mathbb{P}_q(\ell(\zeta,\mathcal{D}) \leqslant \epsilon)$; or one can also exploit different probability inequalities to derive other surrogates, *e.g.*, condition value-at-risk, *i.e.*,

$$\min_q \; KL(q||p) + \lambda \inf_t \left[t + \frac{1}{\epsilon}\mathbb{E}_q[\ell(\zeta,\mathcal{D}) - t]\right]_+, \tag{15}$$

and Bernstein approximation [47]. These surrogates lead to tighter approximation to the chance probability $\mathbb{P}_q(\ell(\zeta) \leqslant \epsilon)$ with the extra cost in optimization.

Remark (parametrization of $q(\zeta)$ and $q(\beta|\zeta)$): We parametrize both $q(\zeta)$ (and the resulting $q(\beta|\zeta)$) as Gaussians with the mean and variance approximated by a multi-layer perceptron (MLP), *i.e.*: $\zeta = \text{MLP}_w(s,a) + \sigma_{w'}\xi,\ \xi \sim \mathcal{N}(0,1)$. w and w' denote the parameters of the MLP.

Remark (connection to Bayesian inference for stochastic processes): Recall the posterior can be viewed as the solution to an optimization [63, 68, 70, 9],

$$q(\zeta|\mathcal{D}) = \operatorname*{argmin}_{q\in\mathcal{P}} \; -\langle q(\zeta), \log p(\zeta,\mathcal{D})\rangle + KL(q(\zeta)||p(\zeta)),$$

Then (13) *mathematically* equivalent to define a log-likelihood $\log p\left(\mathcal{D}|\zeta\right) \propto \ell\left(\zeta, \mathcal{D}\right)$, where $p\left(\mathcal{D}|\zeta\right)$ is a Gibbs point process [15, 65]. For example, plug $f\left(\beta\right) = \frac{1}{2}\beta^\top\beta$ back into (13), we have $\beta^* = \hat{\mathbb{E}}_\mathcal{D}\left[\zeta\left(s,a\right)\cdot\left(\gamma\phi(s',a') - \phi\left(s,a\right)\right)\right] + \left(1-\gamma\right)\mathbb{E}_{\mu_0\pi}\left[\phi\right]$, resulting the optimization

$$\min_q KL\left(q||p\right) + \frac{\lambda}{2\epsilon}\mathbb{E}_q\mathbb{E}_{\mu_0\pi}\hat{\mathbb{E}}_\mathcal{D}[\zeta\left(s_1,a_1\right)^\top k\left(\left(s_1,a_1,s_1',a_1'\right),\left(s_2,a_2,s_2',a_2'\right)\right)\zeta\left(s_2,a_2\right) + 2h\left(s^0,a^0,s,a,s',a'\right)\cdot\zeta\left(s,a\right)], \tag{16}$$

with the kernel $k\left(\left(s_1,a_1,s_1',a_1'\right),\left(s_2,a_2,s_2',a_2'\right)\right) := \left(\gamma\phi(s_1',a_1') - \phi\left(s_1,a_1\right)\right)^\top\left(\gamma\phi(s_2',a_2') - \phi\left(s_2,a_2\right)\right)$ and $h\left(s^0,a^0,s,a,s',a'\right) := \left(1-\gamma\right)\phi\left(s_1^0,a_1^0\right)^\top\left(\gamma\phi(s_2',a_2') - \phi\left(s_2,a_2\right)\right)$. If the prior $p\left(\zeta\right)$ is a $\mathcal{GP}$, the posterior $q\left(\zeta|\mathcal{D}\right)$ will also a $\mathcal{GP}$. Obviously, with different choices of $f^*\left(\cdot\right)$, the BayesDICE framework is far beyond $\mathcal{GP}$.

However, we emphasize although the model define via stochastic processes likelihood in (16) acheives the equivalent optimization, such a likelihood $p\left(\mathcal{D}|\zeta\right)$ is improper in the causality sense as we discussed in Section 3.

Remark (auxilary constraints and undiscounted MDP): As [66] suggested, the non-negative and normalization constraints are important for optimization. We use positive activation functions (ReLU) to ensure the non-negativity of the mean of the $q\left(\zeta\right)$. For the normalization, we consider the chance constraints $\mathbb{P}\left(\left(\hat{\mathbb{E}}_\mathcal{D}\left(\zeta\right)-1\right)^2 \leqslant \epsilon_1\right) \geqslant \xi_1$. By applying the same technique, it leads to an extra term $\frac{\lambda_1}{\epsilon_1}\mathbb{E}_q\left[\max_{\alpha\in\mathbb{R}}\alpha\cdot\hat{\mathbb{E}}_\mathcal{D}\left[\zeta-1\right]\right]$ in (13).

With the normalization condition introduced, the proposed BayesDICE is ready for undiscounted MDP by simply setting $\gamma = 1$ in (13) together with the above extra term for normalization.

C BayesDICE for Exploration vs. Exploitation Tradeoff

In main text, we mainly consider exploiting BayesDICE for estimating various ranking scores for both discounted MDP and undiscounted MDP. In fact, with the posterior of the stationary ratio computed, we can also apply it for better balance between exploration vs. exploitation for policy optimization.

Instead of selecting from a set of policy candidates, the policy optimization is considering all feasible policies and selecting optimistically. Specifically, the feasibility of the stationary state-action distribution can be characterized as

$$\sum_a d\left(s,a\right) = \left(1-\gamma\right)\mu_0 + \mathcal{P}_*d\left(s\right), \quad \forall s\in S, \tag{17}$$

where $\mathcal{P}_*d\left(s\right) := \sum_{\bar{s},\bar{a}} T\left(s|\bar{s},\bar{a}\right)d\left(\bar{s},\bar{a}\right)$. Apply the feature mapping for distribution matching, we obtain the constraint for $\zeta\cdot\pi$ with $\zeta\left(s,a\right) := \frac{d(s)}{d^\mathcal{D}(s,a)}$ as

$$\max_{\beta\in\mathcal{H}_\phi}\ \beta^\top\mathbb{E}_{d^\mathcal{D}}\left[\sum_a\left(\zeta\left(s,a\right)\pi\left(a|s\right)\right)\phi\left(s\right) - \gamma\left(\zeta\left(s,a\right)\pi\left(a|s\right)\right)\phi\left(s'\right)\right] + \left(1-\gamma\right)\mathbb{E}_{\mu_0}\left[\beta^\top\phi\right] - f^*\left(\beta\right) = 0. \tag{18}$$

Then, we have the posteriors for all valid policies should satisfies

$$\lambda\mathbb{P}_q\left(\ell\left(\zeta\cdot\pi,\mathcal{D}\right)\leqslant\epsilon\right)\geqslant\xi, \tag{19}$$

with $\ell\left(\zeta\cdot\pi,\mathcal{D}\right) := \max_{\beta\in\mathcal{H}_\phi}\ \beta^\top\hat{\mathbb{E}}_\mathcal{D}\left[\sum_a\left(\zeta\left(s,a\right)\pi\left(a|s\right)\right)\phi\left(s\right) - \gamma\left(\zeta\left(s,a\right)\pi\left(a|s\right)\right)\phi\left(s'\right)\right] + \left(1-\gamma\right)\mathbb{E}_{\mu_0}\left[\beta^\top\phi\right] - f^*\left(\beta\right)$. Meanwhile, we will select one posterior from among these posteriors of all valid policies optimistically, *i.e.*,

$$\max_{q(\zeta)q(\pi)}\quad \mathbb{E}_q\left[U\left(\tau,r,\mathcal{D}\right)\right] + \lambda_1\xi - \lambda_2 KL\left(q\left(\zeta\right)q\left(\pi\right)||p\left(\zeta,\pi\right)\right) \tag{20}$$

$$\text{s.t.}\quad \mathbb{P}_q\left(\ell\left(\zeta\cdot\pi,\mathcal{D}\right)\leqslant\epsilon\right)\geqslant\xi \tag{21}$$

where $\mathbb{E}_q\left[U\left(\tau,r,\mathcal{D}\right)\right]$ denotes the optimistic policy score to capture the upper bound of the policy value estimation. For example, the most widely used one is

$$\mathbb{E}_q\left[U\left(\tau,r,\mathcal{D}\right)\right] = \mathbb{E}_q\hat{\mathbb{E}}_\mathcal{D}\left[\tau\cdot r\right] + \lambda_u\mathbb{E}_q\left[\left(\hat{\mathbb{E}}_\mathcal{D}\left[\tau\cdot r\right] - \mathbb{E}_q\hat{\mathbb{E}}_\mathcal{D}\left[\tau\cdot r\right]\right)^2\right],$$

where the second term is the empirical variance and usually known as one kind of "exploration bonus".

Then the whole algorithm is iterating between solving (20) and use the obtain policy collecting data into $\mathcal{D}$ in (20).

This Exploration-BayesDICE follows the same philosophy of [49, 48] where the variance of posterior of the policy value is taken into account for exploration. However, there are several significant differences: **i),** the first and most different is the modeling object, [49, 48] is updating with Q-function, while we are handling the dual representation; **ii),** BayesDICE is compatible with arbitary nonlinear function approximator, while [49, 48] considers tabular or linear functions; **iii),** BayesDICE is considering infinite-horizon MDP, while [49, 48] considers fixed finite-horizon case. Therefore, the exploration with BayesDICE pave the path for principle and practical exploration-vs-exploitation algorithm. The regret bound is out of the scope of this paper, and we leave for future work.

D Experiment details and additional discussion and results

D.1 Environments and policies.

Bandit. We create a Bernoulli two-armed bandit with binary rewards where α controls the proportion of optimal arm ($\alpha = 0$ and $\alpha = 1$ means never and always choosing the optimal arm respectively). Our policy selection experiments are based on 5 target policies with $\alpha = [0.75, 0.8, 0.85, 0.9, 0.95]$.

Reacher. We modify the Reacher task to be infinite horizon, and sample trajectories of length 100 in the behavior data. To obtain different behavior and target policies, We first train a deterministic policy from OpenAI Gym [5] until convergence, and define various policies by converting the optimal policy into a Gaussian policy with optimal mean with standard deviation $0.4 - 0.3\alpha$. Our selection experiments are based on 5 target policies with $\alpha = [0.75, 0.8, 0.85, 0.9, 0.95]$.

D.2 Parametrization Details

For the convex function f in (14), we used $f(x) = x^2$. We parametrize the distribution correction ratio as a Gaussian using a deep neural network for the continuous control task. Specifically, we use feed-forward networks with two hidden-layers of 64 neurons each and ReLU as the activation function. The networks are trained using the Adam optimizer ($\beta_1 = 0.99$, $\beta_2 = 0.999$) with batch size 2048 and learning rate 0.0001 on CPUs.

D.3 Additional empirical discussions

BayesDICE v.s. CoinDICE. Because BayesDICE is a Bayesian method, it produces *credible intervals*. While the credible interval is analogous to the frequentist confidence interval, it has different statistical properties, so it is unsurprising that evaluating the credible intervals BayesDICE produces under frequentist measures favors frequentist methods like CoinDICE. The benefit of BayesDICE is its applicability and superior performance for policy selection with arbitrary criteria.

Function approximation in BayesDICE. Constraint embedding can be generalized to use neural network function approximators with potential approximation error. Specifically, as long as the inner product is well-defined, we can characterize the constraints with $max_{f\in\mathcal{F}}\langle f, \Delta\rangle = 0$ where $\mathcal{F}$, i.e., testing function space, can be composed of neural networks. The solution is then known as a "weak solution" in differential equations and finite-element methods. The approximation error induced by such embedding depends on the flexibility of the testing function space. The theoretical analysis considers an idealized scenario which provides guidance. In practice, however, the limited expressibility of the function approximators used, relaxed constraints, and inexact optimization introduce approximation errors, which are challenging to quantify analytically. Empirically, Figure 4 shows that BayesDICE parametrized by kernel and neural network exhibit similar performance, demonstrating the practical effectiveness of neural network as function approximators.

Choice of the prior. The prior of the ratio variables is chosen to be unit Gaussian. We conducted experiments where the prior mean ranges from $[0.1, 10]$ and prior variance ranges from $[0.1, 1]$, and observed the resulting confidence intervals to be similar to those in the paper.

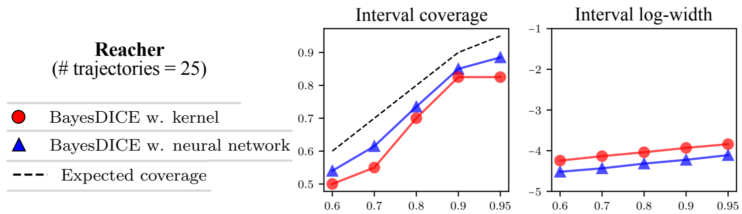

Figure 4: Confidence interval estimation on kernel and neural network parametrized BayesDICE.

Choice of approximate posterior. We chose a Gaussian variational posterior for simplicity. A downside of this choice is that sampled correction ratio can be negative. In practice, we found that is rarely the case, and Gaussian posterior was sufficient to achieve strong performance. Moreover, BayesDICE can naturally incorporate advanced parameterizations, e.g., flow and stochastic differential equations which can ensure positivity.

Comparison to point estimators. The posterior mean estimate of BayesDICE differs from the point estimate in DualDICE due to the prior (i.e., regularization). We summarize the average (across 10 seeds) log RMSE of DualDICE (pt) and of the mean estimate from BayesDICE (μ) on Bandit (B), FrozenLake (F), Taxi (T) and Reacher (R) with varying number of trajectories in the table below. For our choice of prior and these tasks, the performance of the point and mean estimators are similar.

	B50	B100	B200	F50	F100	T20	T50	R25
pt	-4.96	-4.79	-5.69	-9.09	-8.31	-3.36	-5.06	-3.31
μ	-7.86	-9.14	-7.09	-9.94	-9.59	-3.24	-4.11	-3.06

Scalability of BayesDICE. Depending on the evaluation metric chosen, its structural properties can be exploited to nullify the need to test all permutations in Algorithm 1. Such structural properties are present in many natural metrics (such as top-k precision or regret). Therefore, BayesDICE can easily scale to larger numbers of candidate policies.

D.4 Additional experimental results

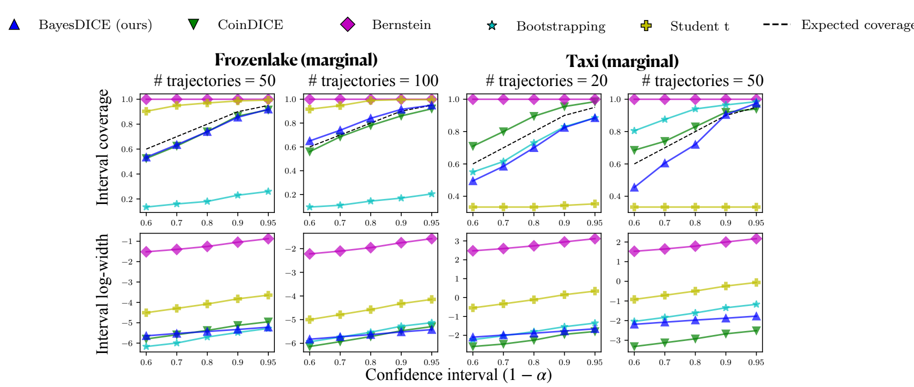

Figure 5: Confidence interval estimation with concentration inequality baselines computed from marginalized importance sampling (as opposed to the per-step importance sampling in the original paper. BayesDICE and CoinDICE still perform much better than methods based on concentration inequality.

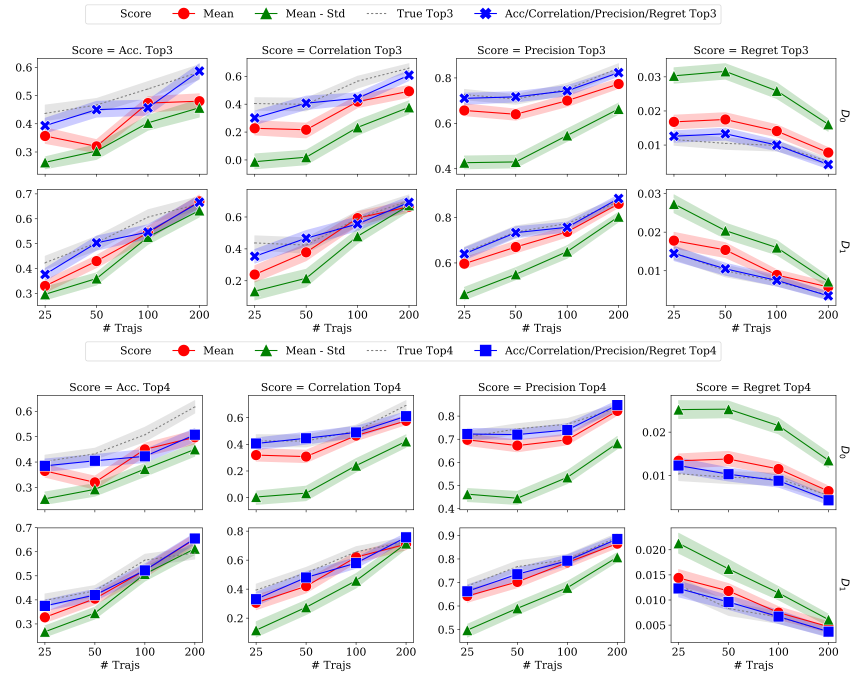

Figure 6: Additional k values for top-k ranking on bandit. Ranking results based on Algorithm 1 (blue lines) always perform better than using mean ("Mean") or high-confidence lower bound ("Mean - Std").

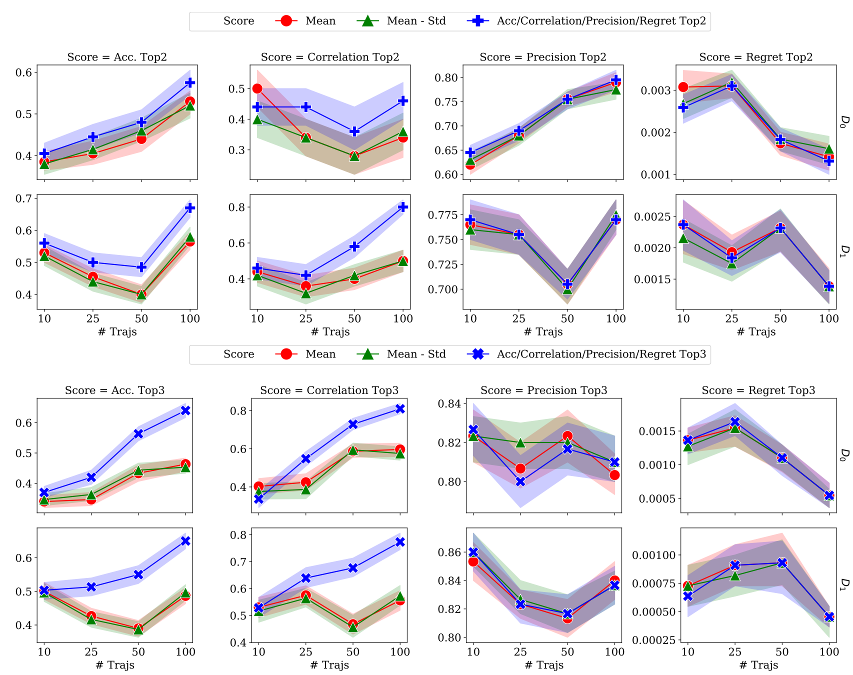

Figure 7: Additional k values for top-k ranking on reacher and additional selection criteria (precision and regret). Ranking results based on Algorithm 1 (blue lines) generally perform much better than using mean ("Mean") or high-confidence lower bound ("Mean - Std") for top-k accuracy and correlation. Precision and regret are similar between posterior samples and the mean/confidence bound based ranking.

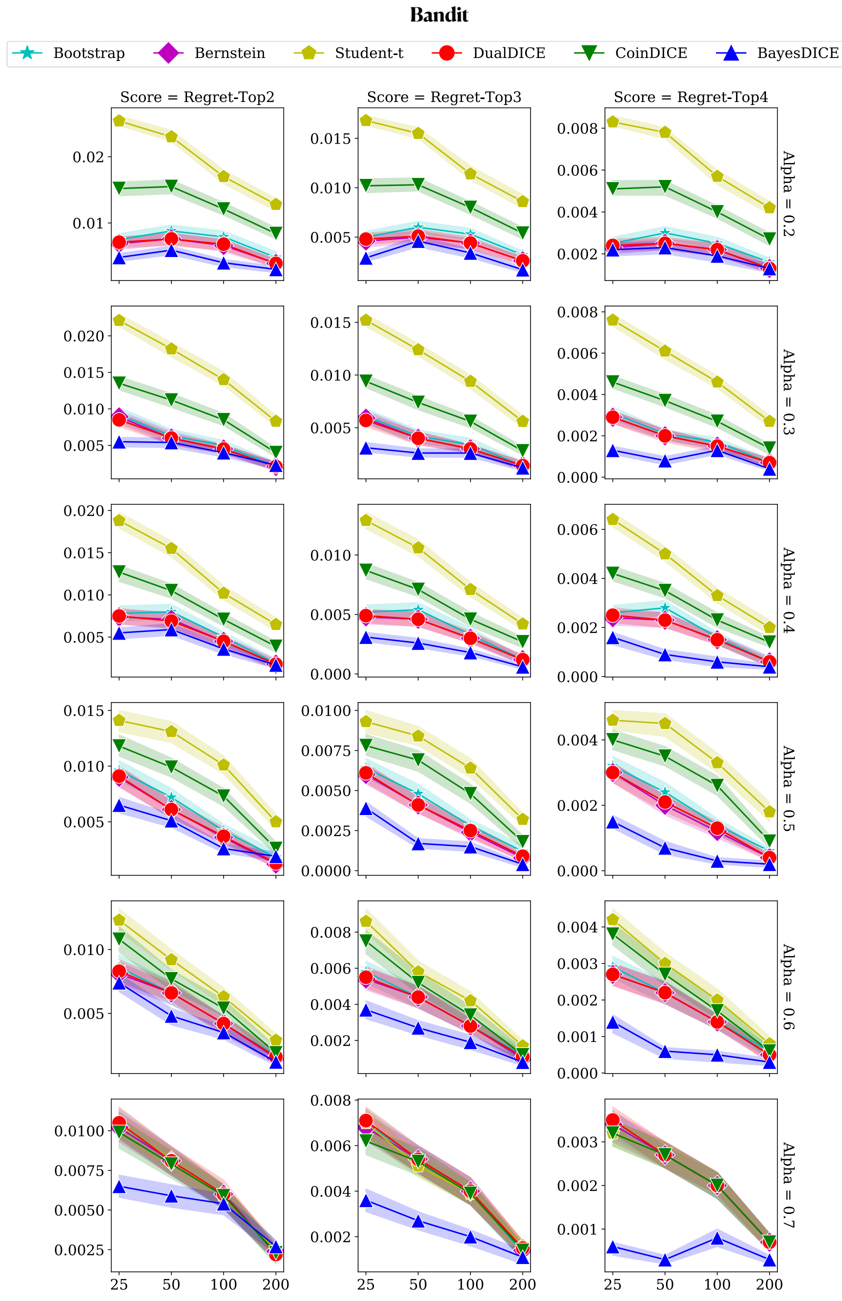

Figure 8: BayesDICE outperforms other confidence-interval based policy selection approaches under the minimum regret criteria across all trajectory lengths, behavior data (higher Alpha means behavior data is closer to optimal policy), and top-k values considered for the bandit task.

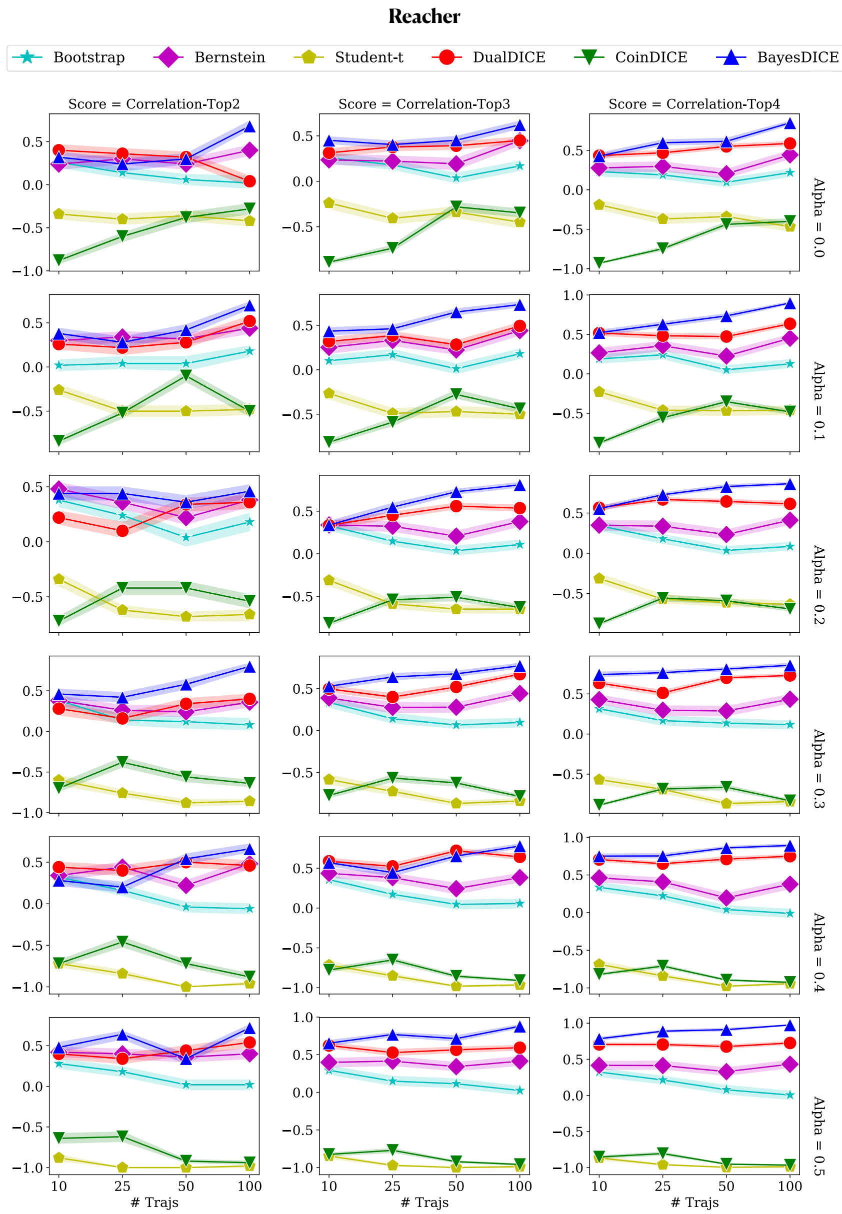

Figure 9: BayesDICE outperforms other confidence-interval based policy selection approaches under the maximum correlation (between true and computed rankings) criteria across all trajectory lengths, behavior data (higher Alpha means behavior data is closer to optimal policy), and top-k values considered for the reacher task.