Sequential Decision Making with Expert Demonstrations under Unobserved Heterogeneity

Anonymous Author(s)
Affiliation
Address
email

Abstract

We study the problem of online sequential decision-making given auxiliary demonstrations from *experts* who made their decisions based on unobserved contextual information. These demonstrations can be viewed as solving related but slightly different tasks than what the learner faces. This setting arises in many application domains, such as self-driving cars, healthcare, and finance, where expert demonstrations are made using contextual information, which is not recorded in the data available to the learning agent. We model the problem as a zero-shot meta-reinforcement learning setting with an unknown task distribution and a Bayesian regret minimization objective, where the unobserved tasks are encoded as parameters with an unknown prior. We propose the Experts-as-Priors algorithm (ExPerior), an empirical Bayes approach that utilizes expert data to establish an informative prior distribution over the learner's decision-making problem. This prior enables the application of any Bayesian approach for online decision-making, such as posterior sampling. We demonstrate that our strategy surpasses existing behaviour cloning and online algorithms, as well as online-offline baselines for multi-armed bandits, Markov decision processes (MDPs), and partially observable MDPs, showcasing the broad reach and utility of ExPerior in using expert demonstrations across different decision-making setups.

1 Introduction

Reinforcement learning (RL) has found success in complex decision-making tasks, spanning areas such as game playing [1, 2, 3], robotics [4, 5], and aligning with human preferences [6]. However, RL's considerable sample inefficiency, necessitating millions of training frames for convergence, remains a significant challenge. A notable body of work within RL has been dedicated to integrating expert demonstrations to accelerate the learning process, employing strategies like offline pretraining [7] and the use of combined offline-online datasets [8, 9]. While these approaches are theoretically sound and empirically validated [10, 11], they typically presume homogeneity between the offline demonstrations and online RL tasks. A vital question arises regarding the effectiveness of these methods when expert data embody heterogeneous tasks, indistinguishable by the learner.

An important example of such heterogeneity is in situations where experts operate with additional information not available to the learner, a scenario previously explored in imitation learning with unobserved contexts [12, 13, 14, 15]. Existing literature either relies on the availability of experts to query during training [16, 17, 18, 19] or focuses on the assumptions that enable imitation learning with unobserved contexts, sidestepping online reward-based interactions [20, 21]. Recent contributions by Hao et al. [22, 23] suggest the utilization of offline expert data for online RL, albeit without accounting for unobserved contextual variations.

Our work addresses the more general challenge of online sequential decision-making given auxiliary offline expert data with *unobserved* heterogeneity. We view such demonstrations as solving related

Submitted to the Automated Reinforcement Learning Workshop at ICML 2024. Do not distribute.

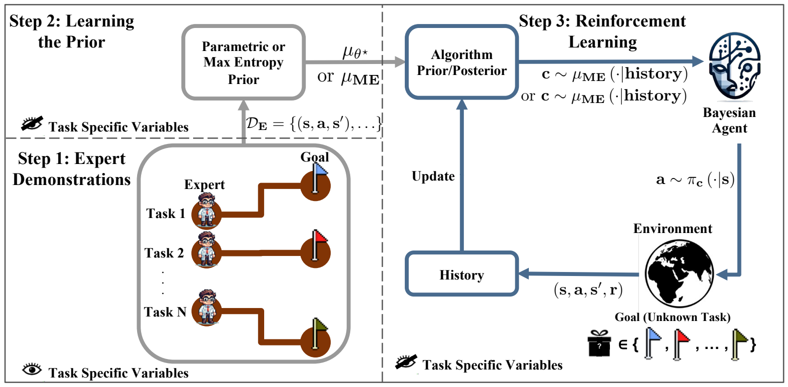

Figure 1: Illustration of ExPerior in a goal-oriented task. Step 1 (Offline): The experts demonstrate their policies for related but different tasks while observing the goal type. Step 2 (Offline): The expert data $\mathcal{D}_{\mathrm{E}}$ only contains the trajectories states/actions — goal types are not collected. We form a parametric or nonparametric max-entropy prior distribution over tasks using $\mathcal{D}_{\mathrm{E}}$. Step 3 (Online): The goal type is unknown but drawn from the same distribution of goals in Step 1. The learner uses the learned prior for posterior sampling.

yet distinct tasks from those faced by the learner, where differences remain invisible to the learner. For instance, in a personalized education scenario, while a learning agent might access observable characteristics like grades or demographics, it might remain oblivious to factors such as learning styles, which are visible to an expert teacher and can significantly influence teaching methods. A naïve imitation learning algorithm without access to this "private" information will only learn a single policy for each observed characteristic [24], leading to sub-optimal decisions. On the other hand, a purely online approach will require extensive trial and error to result in meaningful decisions.

We integrate offline expert data with online RL, treating the scenario as a zero-shot meta-reinforcement learning (meta-RL) problem with an unknown distribution over tasks (unobserved factors). Unlike typical meta-RL frameworks where the learner is exposed to multiple tasks during training (different students in our example) to learn the underlying task distribution [25, 26].

Contributions: We define a Bayesian regret minimization objective and consider different tasks as parameters under an unknown prior distribution. We use empirical Bayes to derive an informative prior over the decision-making task from expert data. We use the learned prior distribution to drive exploration in the online RL task, using approaches like posterior sampling [27]. We propose two procedures to learn such a prior: (1) a parametric approach that can utilize any existing knowledge about the parametric form of the prior distribution, and (2) a nonparametric approach that employs the principle of maximum entropy when such prior knowledge does not exist. We call our framework Experts-as-Priors or ExPerior for short (see Figure 1). ExPerior outperforms existing offline, online, and offline-online baselines in multi-armed bandits, Markov decision processes (MDPs), and partially observable MDPs. For multi-armed bandits, we find the Bayesian regret incurred by ExPerior is proportional to the entropy of the optimal action under the prior distribution, aligning with the entropy of expert policy if the experts are optimal. We introduce a frequentist algorithm for multi-armed bandits and prove a Bayesian regret bound proportional to a term that closely resembles the entropy of the optimal action. Our results suggest using the entropy of expert demonstrations to evaluate the impact of unobserved factors.

2 Related Work

Our work is an addition to the recent body of reinforcement learning research that leverages offline demonstrations to speed up online learning [28, 10, 29, 7, 9]. Classic algorithms such as DDPGfD [30] and DQfD [31] achieve this by combining imitation learning and RL. They modify DDPG [5] and DQN [1] by warm-starting the algorithms' replay buffers with expert trajectories and ensuring that the offline data never gets overridden by online trajectories. Closely related to our study is the meta-RL literature, which aims to accelerate learning in a given RL task by using prior experience from related tasks [32, 33, 34]. These papers present model-agnostic meta-learning training objectives to maximize the expected reward from novel tasks as efficiently as possible.

Two unique features distinguish our problem from the settings considered above. First, our setting assumes heterogeneity within the offline data and with the online RL task that is unobserved to the learner, while the (optimal) experts are privy to that heterogeneity. Second, we assume the learner will only interact with one online task, making our setup similar to zero-shot meta-RL [35, 36, 37]. Most similar to our work is the ExPLORe algorithm [38], which assigns optimistic rewards to the offline data during the online interaction and runs an off-policy algorithm using both online and labelled offline data as buffers. For our setting, the algorithm incentivizes the learner to explore the expert trajectories, leading to faster convergence. We consider this work one of our baselines.

Our methodology utilizes only the state-action trajectory data from expert demonstrations without task-specific information or reward labels. Other similar methods require additional offline information. For example, Nair et al. [29] assume that the offline data contains the reward labels and use that to pre-train a policy, which is then fine-tuned online. Mendonca et al. [39] require task labelling for each trajectory and use the offline data to learn a single meta-learner. Similarly, Zhou et al. [40] and Rakelly et al. [41] require the task label and reward labels. They then infer the task during online interaction and use the task-specific offline data. Finally, our methodology builds on posterior sampling [42]. Hao et al. [22, 23] consider a similar problem using posterior sampling to leverage offline expert demonstration data to improve online RL. However, they assume homogeneity between the expert data and online tasks. In contrast, our setting accounts for heterogeneity.

3 Problem Setup

Decision Model for Unobserved Heterogeneity of Tasks. To account for unobserved heterogeneity, we consider a generalization of finite-horizon Markov Decision Processes (MDPs) with a notion of probabilistic task variables [43, 13, 21]. The MDP's underlying model will additionally depend on an unobserved task variable that encapsulates some information about the specific task. In a personalized education setup where teaching a student corresponds to a task, and the learning agent can observe students' characteristics, like their demographic status and grades. Other factors, such as the student's learning style (e.g., visual learners vs self-study), may not be readily available, even though they are important in determining the optimal teaching style.

Let $\mathcal{C}$ be the set of all *unobserved* variables that can describe the heterogeneity of potential tasks (e.g., the set of all possible learning styles). A (contextual) MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, R, H, \rho, \mu^\star)$ is parameterized by states $\mathcal{S}$, actions $\mathcal{A}$, transition function $\mathcal{T} : \mathcal{S} \times \mathcal{A} \times \mathcal{C} \to \Delta(\mathcal{S})$, reward function $R : \mathcal{S} \times \mathcal{A} \times \mathcal{C} \to \Delta(\mathbb{R})$, horizon $H > 0$, initial state distribution $\rho \in \Delta(\mathcal{S})$, and task distribution $\mu^\star$. We assume the transition/reward functions and $\mu^\star$ are unknown, and for simplicity, ρ does not depend on the task variable. For each task $c \sim \mu^\star$, we consider T episodes, where at the beginning of each episode $t \in [T]$, an initial state $s_1 \sim \rho$ is sampled. Then, at each timestep $h \in [H]$, the learner chooses an action $a_h \in \mathcal{A}$, observes a reward $r_h \sim R(s_h, a_h, c)$ and the next state $s_{h+1} \sim \mathcal{T}(s_h, a_h, c)$. Without loss of generality, we assume the states are partitioned by $[H]$ to make the notation invariant to timestep h. Let Π be the set of all Markovian policies. For a policy function $\pi : \mathcal{S} \to \Delta(\mathcal{A}) \in \Pi$ and task variable c, we define the value function $V_c(\pi) = \mathbb{E}\left[\sum_{h=1}^{H} r_h \,\middle|\, \pi, c\right]$ and the Q-function as $Q_c^\pi(s, a) := \mathbb{E}\left[\sum_{h'=h}^{H} r_{h'} \,\middle|\, s_h = s, a_h = a, \pi, c\right]$ for all $s \in \mathcal{S}, a \in \mathcal{A}$. Moreover, we define the optimal policy for a task variable $c \in \mathcal{C}$ as $\pi_c := \arg\max_{\pi \in \Pi} V_c(\pi)$. Note that since the task variable is unobserved, the learner's policy will not depend on it. The learning agent's goal is to learn history-dependent distributions $p^1, \ldots, p^T \in \Delta(\Pi)$ over Markovian policies to minimize the expected regret, defined as $\mathrm{Reg} := \mathbb{E}_{c\sim\mu^\star}\left[\sum_{t=1}^{T} V_c(\pi_c) - \mathbb{E}_{\pi^t\sim p^t}\left[V_c(\pi^t)\right]\right]$.

The above setup assumes a fixed distribution $\mu^\star$ over the set of learning styles and aims to minimize expected regret over the population of students. Our setup and regret assume the unobserved factors remain fixed during training. This captures scenarios wherein the unobserved variables correspond to less-variant factors (a student's learning style is more likely to remain unchanged). No learning algorithm can control the regret value if we allow the unobserved factors to change arbitrarily throughout T episodes without access to hidden information. Consider a two-armed bandit with a task value drawn with uniform probability from $\mathcal{C} = \{c_1, c_2\}$ and can change at each episode. Assume the expected reward of the first arm under c_1 and c_2 is one and zero, respectively, and it is reversed for the other arm. Any algorithm that does not have access to c would result in linear regret since each action is sub-optimal with a probability of 0.5, independent of the algorithm's choice.

Remark. Our setup can be formulated as a Bayesian model parameterized by $\mathcal{C}$, and our regret can be seen as the Bayesian regret of the learner. However, the distribution $\mu^\star$ is not the learner's prior belief about the true model as it is often formulated in Bayesian learning, but a distribution over potential tasks that the learner can encounter. Our setup can thus be seen as a meta-learning problem. In fact, it is *zero-shot* meta-learning since we do not assume having access to more than one online task during training — we only learn the prior distribution using the offline data.

Expert Demonstrations. In addition to the online setting described above, we assume the learner has access to an offline dataset of expert demonstrations $\mathcal{D}_{\mathrm{E}}$, where each demonstration $\tau_{\mathrm{E}} = (s_1, a_1, s_2, a_2, \ldots, s_H, a_H, s_{H+1})$ refers to an interaction of the expert with a decision-making task during a *single* episode, containing the actions made by the expert and the resulting states. We assume that the unobserved task-specific variables for $\mathcal{D}_{\mathrm{E}}$ are drawn i.i.d. from distribution $\mu^\star$, and the expert had access to such unobserved variables (private information) during their decision-making. Moreover, we assume the expert follows a near-optimal strategy [22, 23].

Assumption 1 (Noisily Rational Expert)**.** For any $c \in \mathcal{C}$, experts select actions based on a distribution defined as $p_{\mathrm{E}}(a \mid s\,;\, c) \propto \exp\{\beta \cdot Q_c^{\pi_c}(s, a)\}$, for all $s \in \mathcal{S}, a \in \mathcal{A}$, and some known competence value of $\beta \in [0, \infty]$. In particular, the expert follows the optimal policy if $\beta \to \infty$.

We assume experts do not provide any rationale for their strategy, nor do we have access to rewards in the offline data; this is a combination of imitation and online learning rather than offline RL.

4 Experts-as-Priors Framework for Unobserved Heterogeneity

Our goal is to leverage offline data to help guide the learner through its interaction with the decision-making task. The key idea is to use expert demonstrations to infer a *prior* distribution over $\mathcal{C}$ and then to use a Bayesian approach such as posterior sampling [27] to utilize the inferred prior for a more informative exploration. If the current task is from the same distribution of tasks in the offline data, we expect that using such priors will lead to faster convergence to optimal trajectories compared to the commonly used non-informative priors. Consider the personalized education example. Suppose we have gathered offline data on an expert's teaching strategies for students with similar observed information like grade, age, location, etc. The teacher can observe more fine-grained information about the students that is generally absent from the collected data (e.g., their learning style). Our work relies on the following observation: The space of the optimal strategies for students with similar observed information but different learning styles is often much smaller than the space of all possible strategies. With the inferred prior distribution, the learner needs only to focus on the span of potentially optimal strategies for a new student, allowing for significantly more efficient exploration.

We resort to empirical Bayes and use maximum marginal likelihood estimation [44] to construct a prior distribution from $\mathcal{D}_{\mathrm{E}}$. Given a probability distribution (prior) μ on $\mathcal{C}$, the marginal likelihood of an expert demonstration $\tau_{\mathrm{E}} = (s_1, a_1, s_2, a_2, \ldots, s_H, a_H, s_{H+1}) \in \mathcal{D}_{\mathrm{E}}$ is given by

$$\mathrm{P}_{\mathrm{E}}(\tau_{\mathrm{E}}\,;\,\mu) = \mathbb{E}_{c\sim\mu}\left[\rho(s_1) \cdot \prod_{h=1}^{H} p_{\mathrm{E}}(a_h \mid s_h\,;\, c)\, \mathcal{T}(s_{h+1} \mid s_h, a_h, c)\right]. \tag{1}$$

We aim to find a prior distribution to maximize the log-likelihood of $\mathcal{D}_{\mathrm{E}}$ under the model described in (1). This is equivalent to minimizing the KL divergence between the marginal likelihood P_{E} and the empirical distribution of expert demonstrations, which we denote by $\widehat{\mathrm{P}}_{\mathrm{E}}$. In particular, we form an uncertainty set over the set of plausible priors as $\mathcal{P}(\epsilon) := \left\{\mu\,;\, \mathrm{D}_{\mathrm{KL}}\left(\widehat{\mathrm{P}}_{\mathrm{E}} \,\middle\|\, \mathrm{P}_{\mathrm{E}}(\cdot\,;\,\mu)\right) \le \epsilon\right\}$, where the value of ϵ can be chosen based on the number of samples so the uncertainty set contains the true prior with high probability [35]. However, the set of plausible priors does not uniquely identify the appropriate prior. In fact, even for $\epsilon = 0$, $\mathcal{P}(\epsilon)$ can have infinite plausible priors. To solve this ill-posed problem, we propose two approaches, parametric and nonparametric prior learning.

Parametric Experts-as-Priors. For settings where we have existing knowledge about the parametric form of the prior, we can directly apply maximum marginal likelihood estimation to learn it. In particular, we define the parametric expert prior as the following. Note that we can calculate the gradients of the marginal likelihood using the score function estimator [45].

Definition 1 (Parametric Expert Prior)**.** Let $\boldsymbol{\Theta}$ be a set of plausible prior distribution parameters (e.g., Beta distribution parameters for a Bernoulli bandit). We call $\mu_{\boldsymbol{\theta}^\star}$ a parametric expert prior, iff $\theta^\star \in \arg\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \sum_{\tau\in\mathcal{D}_{\mathrm{E}}} -\log \mathrm{P}_{\mathrm{E}}(\tau\,;\,\mu_{\boldsymbol{\theta}})$.

Nonparametric Experts-as-Priors. For settings where there is no existing knowledge on the parametric form of the prior, we can employ the principle of maximum entropy to choose the *least informative* prior that is compatible with expert data:

Definition 2 (Max-Entropy Expert Prior)**.** Let μ_0 be a non-informative prior on $\mathcal{C}$ (e.g., a uniform distribution). Given some $\epsilon > 0$, we define the maximum entropy expert prior μ_{ME} as the solution to the following optimization problem:

$$\mu_{\text{ME}} = \arg\min_{\mu} \ \mathrm{D_{KL}}\left(\mu \,\|\, \mu_0\right) \quad \text{s.t.} \quad \mu \in \mathcal{P}(\epsilon). \tag{2}$$

Note that the set of plausible priors $\mathcal{P}(\epsilon)$ is a convex set, and therefore, (2) is a convex optimization problem. We derive the solution to problem (2) using Fenchel's duality theorem [46, 47]:

Proposition 1 (Max-Entropy Expert Prior)**.** *Let $N = |\mathcal{D}_E|$ be the number of demonstrations in $\mathcal{D}_E$. For each $c \in \mathcal{C}$ and demonstration $\tau_E = (s_1, a_1, s_2, a_2, \dots, s_H, a_H, s_{H+1}) \in \mathcal{D}_E$, define $m_{\tau_E}(c)$ as the (partial) likelihood of τ_E under c, i.e., $m_{\tau_E}(c) = \prod_{h=1}^{H} p_E\left(a_h \mid s_h\,;\, c\right) \mathcal{T}\left(s_{h+1} \mid s_h, a_h, c\right)$.*

Denote $\mathbf{m}(c) \in \mathbb{R}^N$ as the vector with elements $m_{\tau_E}(c)$ for $\tau_E \in \mathcal{D}_E$. Moreover, let $\lambda^\star \in \mathbb{R}^{\geq 0}$ be the optimal solution to the Lagrange dual problem of (2). Then, the solution to optimization (2) is:

$$\mu_{ME}(c) = \lim_{n\to\infty} \frac{\exp\left\{\mathbf{m}(c)^\top \boldsymbol{\alpha}_n\right\}}{\mathbb{E}_{c\sim\mu_0}\left[\exp\left\{\mathbf{m}(c)^\top \boldsymbol{\alpha}_n\right\}\right]},$$

where $\{\boldsymbol{\alpha}_n\}_{n=1}^{\infty}$ is a sequence converging to the following supremum:

$$\sup_{\boldsymbol{\alpha}\in\mathbb{R}^N} -\log \mathbb{E}_{c\sim\mu_0}\left[\exp\left\{\mathbf{m}(c)^\top \boldsymbol{\alpha}\right\}\right] + \frac{\lambda^\star}{N}\sum_{i=1}^{N} \log\left(\frac{N\cdot\alpha_i}{\lambda^\star}\right). \tag{3}$$

The proof is provided in Appendix A.3. Instead of solving for $\lambda^\star$, we set it as a hyperparameter and then solve (3). Even though Proposition 1 requires the correct form of Q-functions for different values of c, we will see in the following sections that we can parameterize the Q-functions and treat those parameters as a proxy for the unobserved factors. Once such a prior is derived, we can employ any Bayesian approach for the decision-making task. We provide a pseudo-algorithm for ExPerior in Appendix B. The following sections will detail the algorithm for bandits and MDPs.

5 Learning in Bandits

K-armed Bandits. For K-armed bandits, note that $\mathcal{S} = \emptyset$, $H = 1$, and $\mathcal{A} = \{1, \dots, K\}$. Each expert demonstration $\tau_{\text{E}} = a$ will be the pulled arm by the expert for a particular bandit, and the (partial) likelihood function in Proposition 1 can be simplified as $m_{\tau_{\text{E}}}(c) = p_{\text{E}}\left(a\,;\, c\right)$. This likelihood function only depends on the task variable c through the expert policy p_{E}, and since p_{E} only depends on c through the mean reward function (Assumption 1), we can consider the set of mean reward functions as a proxy for the unobserved task variables $\mathcal{C}$. e.g. in a Bernoulli K-armed bandit setting, we can define $\mathcal{C}_{\text{Ber}} = \left\{a \mapsto \langle \mathbf{e}_a, \boldsymbol{\vartheta}\rangle \,;\, \boldsymbol{\vartheta} \in [0,1]^K\right\}$.

Stochastic Contextual Bandits. In contextual bandits, the state space $\mathcal{S}$ is the set of contexts and $H = 1$. Therefore, the likelihood function for a demonstration $\tau_{\text{E}} = (s, a)$ will be $m_{\tau_{\text{E}}}(c) = p_{\text{E}}\left(a \mid s\,;\, c\right)$. Like K-armed bandits, the likelihood function only depends on c through the expert policy. Therefore, we can similarly define the set of mean reward functions as the proxy for the unobserved task variables. For instance, we can consider the task parameters for linear contextual bandits as $\mathcal{C}_{\text{Lin}} = \left\{(s, a) \mapsto \langle \phi\left(s, a\right), \boldsymbol{\vartheta}\rangle \,;\, \boldsymbol{\vartheta} \in \mathbb{R}^d\right\}$, for a known feature function $\phi : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^d$.

Posterior Sampling. With the above parameterizations of $\mathcal{C}$, we can use Proposition 1 to derive the maximum entropy prior distribution over the task parameters. However, we cannot sample from the exact posterior since the derived prior is not a conjugate prior for standard likelihood functions. Instead, we resort to approximate posterior sampling via stochastic gradient Langevin dynamics (SGLD) [48]. We call this method `ExPerior-MaxEnt` in our experiments. We also employ a parametric approach as discussed in section 4, which we call `ExPerior-Param`. In particular, we use the Beta distribution as our prior model and learn the parametric expert prior in Definition 1. `ExPerior-Param` has an advantage over `ExPerior-MaxEnt` since it provides exact posterior sampling for Bernoulli bandits.

We aim to evaluate our approach compared to other baselines, including online methods that do not use expert data and offline behaviour cloning. We provide an empirical regret analysis for ExPerior based on the informativeness of expert data, number of actions, and number of training episodes. We also discuss the robustness of ExPerior to misspecified expert models and the advantage of `ExPerior-MaxEnt` to `ExPerior-Param` when the parametric prior model is misspecified. To characterize the effect of expert data on the learner's performance, we propose an alternative for K-armed bandits inspired by the successive elimination and derive a Bayesian regret bound for it.

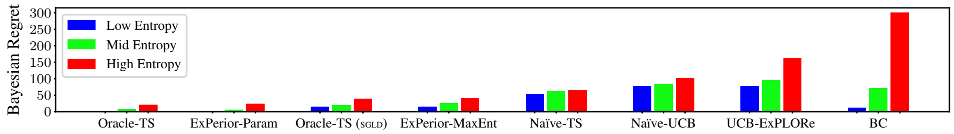

Figure 2: The Bayesian regret of ExPerior and baselines for K-armed Bernoulli bandits ($K = 10$). We consider three categories of task distributions based on the entropy of the optimal action.

Experiments. We consider K-armed Bernoulli bandits for our experimental setup (code: `https://anonymous.4open.science/r/ExPerior-0773`). We evaluate the learning algorithms in terms of the Bayesian regret over multiple task distributions $\mu^\star$. We consider up to $N_{\mu^\star} = 64$ different beta task distributions, where their parameters are chosen to span a different range of heterogeneity, consisting of tasks with various expert data informativeness. To estimate the Bayesian regret, we sample $N_{\text{task}} = 128$ bandit tasks from each task distribution and calculate the average regret. We use $N_{\text{E}} = 1000$ expert demonstrations for each task distribution in our experiments. We compare ExPerior to the following baselines: (1) Behaviour cloning (`BC`), which learns a policy by minimizing the cross-entropy loss between the expert demonstrations and the agent's policy solely based on offline data. (2) Naïve Thompson sampling (`Naïve-TS`) that chooses the action with the highest sampled mean from a posterior distribution under an uninformative prior. (3) Naïve upper confidence bound (`Naïve-UCB`) algorithm that selects the action with the highest upper confidence bound. Both `Naïve-TS` and `Naïve-UCB` ignore expert demonstrations. (4) `UCB-ExPLORe`, a variant of the algorithm proposed by Li et al. [38] tailored to bandits. It labels the expert data with optimistic rewards and then uses it alongside online data to compute the upper confidence bounds for exploration, and (5) `Oracle-TS`, which performs exact Thompson sampling having access to the true task distribution $\mu^\star$. For a more fair comparison, we also consider a variant of `Oracle-TS`, which uses SGLD for approximate posterior sampling.

Comparison to baselines. Figure 2 demonstrates the average Bayesian regret for various task distributions over $T = 1500$ episodes with $K = 10$ arms. To better understand the effect of expert data, we categorize the task distributions by the entropy of their optimal actions into low entropy (less than 0.8), high entropy (greater than 1.6), and medium entropy. `Oracle-TS` and `ExPerior-Param` outperform other baselines, yet the performance of ExPerior is comparable to the SGLD variant of `Oracle-TS`. This indicates that the maximum entropy prior derived from Proposition 1 closely approximates the true task distribution, $\mu^\star$, with the performance difference with `Oracle-TS` is primarily due to approximate posterior sampling. Moreover, the pure online algorithms `Naïve-TS` and `Naïve-UCB`, which disregard expert data, display similar performance across different entropy levels, contrasting with other algorithms that show significantly reduced regret in low-entropy contexts. This underlines the impact of expert data in settings where the unobserved confounding has less effect on the optimal actions. Specifically, in the extreme case of no task heterogeneity, `BC` is anticipated to yield optimal performance. Additionally, `Naïve-UCB` surpasses `UCB-ExPLORe` in medium and high entropy settings, possibly due to the over-optimism of the reward labelling in Li et al. [38], which can hurt the performance when the expert demonstrations are uninformative.

Empirical regret analysis for Experts-as-Priors. We examine how the quality of expert demonstrations affects the Bayesian regret achieved by Algorithm 2. Settings with highly informative demonstrations, where unobserved factors minimally affect the optimal action, should exhibit near-zero regret since there is no diversity in the tasks, and the experts are near-optimal. Conversely, in scenarios where unobserved factors significantly influence the optimal actions, we anticipate the regret to align with standard online regret bounds, similar to the outcomes of Thompson sampling with a non-informative prior. We conduct trials with ExPerior and `Oracle-TS` across various numbers of arms over $T = 1500$ episodes, calculating the mean and standard error of Bayesian regret across distinct task distributions. As depicted in Figure 3 (a), both ExPerior and `Oracle-TS` yield sub-linear regret relative to K and T, comparable to the established regret bound of $\mathcal{O}(\sqrt{KT})$ for Thompson sampling. However, the middle panel indicates that the regret of ExPerior is proportional to the entropy of the optimal action, having an almost *linear* relationship. This observation seems to be in contrast with the standard Bayesian regret bounds for Thompson sampling under correct prior that have shown a sublinear relationship of $\mathcal{O}\left(\sqrt{\text{Ent}(\pi_c)}\right)$, where $\text{Ent}(\pi_c)$ denotes the entropy of the optimal action under $\mu^\star$ [49]. We analyze this observation more concretely below.

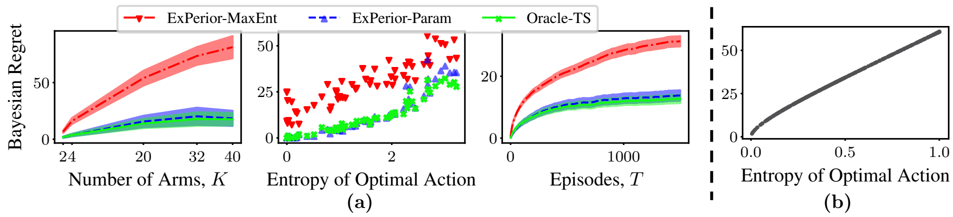

Figure 3: (a) Empirical analysis of ExPerior's regret in Bernoulli bandits based on the (left) number of arms, (middle) entropy of the optimal action, and (right) number of episodes. (b) The regret bound from Theorem 2 V.S. the entropy of the optimal action. The linear relationship is consistent with the middle panel of (a).

Ablations. We run additional experiments in Appendix C.1 to assess the robustness of ExPerior to misspecified experts. We create expert data from different experts with various competence levels, such as optimal, noisily rational, and random-optimal experts, where the latter chooses an action optimally with a fixed probability and randomly otherwise. Table 2 in the appendix shows ExPerior's robustness to different expert models. With $\beta = 10$ for training `ExPerior-MaxEnt` and $\beta = 1$ for `ExPerior-Param` achieves consistent out-performance among different expert types. We evaluate the advantage of learning nonparametric max-entropy prior over misspecified parametric priors in Table 3. Even though `ExPerior-Param` with Beta model outperforms `ExPerior-MaxEnt`, `ExPerior-MaxEnt` is superior to `ExPerior-Param` if the prior is chosen as Gaussian or Gamma.

An Alternative Frequentist Approach for K-armed Bandits To analyze the effect of expert data on the Bayesian regret, we devise an alternative *frequentist* approach, based on the successive elimination algorithm [50], which follows a similar intuition to Experts-as-Priors. In particular, we prove a bound on its Bayesian regret and show that the derived bound is proportional to a term that closely resembles the entropy of the optimal action, showing that the observation in the middle panel of Figure 3 (a) is consistent within different approaches.

The idea of successive elimination is to identify suboptimal arms and deactivate them over time. In particular, it runs a uniform sampling policy among active arms and builds confidence intervals for each. It then deactivates all the arms with an upper confidence bound smaller than at least one arm's lower confidence bound. We modify this algorithm using the policy derived from expert demonstrations instead of a uniform sampling policy. Recall that in K-armed bandits, each expert trajectory τ_{E} represents the pulled arm by the expert. Hence, the empirical distribution of expert demonstrations can be seen as a sampling policy over different arms. The concrete algorithm is provided in Algorithm 1 in Appendix A.4. We now provide a Bayesian regret bound of this algorithm.

Theorem 2. *Consider a stochastic K-armed bandit and let p be the empirical expert policy. Assume that (i) the mean reward function is bounded in $[0,1]$ for all arms, (ii) $T \geq \frac{1}{\min_{a;p(a)\neq 0} p(a)}$, (iii) the expert is optimal, i.e., $\forall a \in \mathcal{A}: \; p(a) = \mathrm{P}_E(a\,;\mu^\star)$ and $\beta \to \infty$, and (iv) the learner follows Algorithm 1. Then, with probability at least $1-\delta$,*

$$\mathit{Reg} \lesssim \sqrt{T \log\left(TK/\delta\right)} \sum_{a,a' \in \mathcal{A}, a \neq a'} \sqrt{\frac{p(a)}{p(a)+p(a')}\left(1 - \frac{p(a)}{p(a)+p(a')}\right)} \left[\sqrt{p(a)} + \sqrt{p(a')}\right]. \quad (4)$$

See Appendix A.4 for the proof. Two terms in (4) depend on expert data: (1) The relative standard deviation between any two pairs of arms and (2) a scaling factor that depends on the magnitude of probability that the arms are optimal. For homogeneous demonstrations, where the expert data only includes one unique pulled arm, the standard deviation (Term 1) is zero, resulting in zero regret. On the other hand, in extreme heterogeneity, where the empirical expert distribution is uniform over the arms, we have Reg $\lesssim \sqrt{KT \log T}$, a similar bound for standard successive elimination. Finally, to assess the relationship between the regret bound and the entropy of the expert data, we fix $K = 2$, $T = 100$, and plot the bound from (4) as a function of the entropy of the optimal action for various task distributions. Figure 3 (b) demonstrates a linear relationship, similar to the regret incurred by ExPerior in Figure 3 (a). This observation opens up new directions to further analyze the theoretical regret for ExPerior and propose similar frequentist approaches for MDPs.

6 Learning in Markov Decision Processes (MDPs)

For MDPs, we need to parameterize both the mean reward and transition functions. However, we assume the transition functions are invariant to the task variables to simplify our methodology and

Table 1: The average reward per episode in Frozen Lake (PODMP) after 90,000 training steps.

	Fixed # Hazard = 9				Fixed $\beta = 1$			
	$\beta = 0.1$	$\beta = 1$	$\beta = 2.5$	$\beta = 10$	# Hazard = 2	# Hazard = 5	# Hazard = 7	# Hazard = 9
ExPerior-MaxEnt	-22.58 $\pm$ 1.17	**6.00 $\pm$ 0.00**	3.58 $\pm$ 0.89	1.62 $\pm$ 1.85	11.47 $\pm$ 0.52	**5.71 $\pm$ 0.67**	**6.00 $\pm$ 0.00**	**6.00 $\pm$ 0.00**
ExPerior-Param	-23.32 $\pm$ 0.69	-4.31 $\pm$ 1.80	5.27 $\pm$ 0.51	**6.00 $\pm$ 0.00**	**12.00 $\pm$ 0.37**	2.11 $\pm$ 1.41	5.42 $\pm$ 0.40	-4.31 $\pm$ 1.80
Naïve Boot-DQN	-23.32 $\pm$ 0.69	-23.32 $\pm$ 0.69	-23.32 $\pm$ 0.69	-23.32 $\pm$ 0.69	-14.36 $\pm$ 5.88	-20.57 $\pm$ 2.91	-20.39 $\pm$ 1.75	-23.32 $\pm$ 0.69
ExPLORe	**5.99 $\pm$ 0.00**	**6.00 $\pm$ 0.00**	**6.00 $\pm$ 0.00**	**6.00 $\pm$ 0.00**	-30.68 $\pm$ 12.40	-10.64 $\pm$ 16.64	-13.00 $\pm$ 19.00	**6.00 $\pm$ 0.00**
Optimal	6.00 $\pm$ 0.00	6.00 $\pm$ 0.00	6.00 $\pm$ 0.00	6.00 $\pm$ 0.00	12.00 $\pm$ 0.37	6.53 $\pm$ 0.31	6.00 $\pm$ 0.00	6.00 $\pm$ 0.00

avoid extra modelling assumptions. Under this assumption, it is sufficient to parameterize the *optimal* Q-functions, e.g., using a deep Q-network (DQN) and treat those parameters as a proxy for the task variables, i.e., $\mathcal{C}_{\text{MDP}} := \{(s, a) \mapsto Q\,(s, a\,;\,\boldsymbol{\theta})\ ;\ \boldsymbol{\theta} \in \boldsymbol{\Theta}\}$, where $\boldsymbol{\Theta}$ is the set of parameters for a DQN. We can then derive a closed-form log-pdf of the posterior distribution under the maximum entropy prior. See Appendix A.5 for details. The derived posterior log-pdf can then be used as the loss function for DQN Langevin Monte Carlo [51, 52] as the counterpart for Thompson sampling with SGLD. However, running Langevin dynamics can lead to highly unstable policies due to the complexity of the optimization landscape in DQNs. Instead of sampling from the posterior distribution, we use a heuristic that combines the learned prior distribution with bootstrapped DQNs [53].

The original method of Bootstrapped DQNs utilizes an ensemble of L randomly initialized Q-networks. It samples a Q-network uniformly at each episode and uses it to collect data. Then, each Q-network is trained using the temporal difference loss on parts of or possibly the entire collected data. This method and its subsequent iterations [54, 55, 56] achieve deep exploration by ensuring diversity among the learned Q-networks. To incorporate Bootstrapped DQN into the ExPerior framework and utilize the expert data, we can formulate the ensemble as a discrete prior distribution over the Q-networks. Let $\boldsymbol{\theta}_{\text{ens}} = \left(\boldsymbol{\theta}^1_{\text{ens}}, \ldots, \boldsymbol{\theta}^L_{\text{ens}}\right)$ be the parameter vector for an ensemble of Q-functions. We can define the ensemble prior, parameterized by $\boldsymbol{\theta}_{\text{ens}}$, as $\mu_{\boldsymbol{\theta}_{\text{ens}}}(\boldsymbol{\theta}) := \frac{1}{L}\sum_{i=1}^{L} \mathbb{I}\left(\boldsymbol{\theta}^i_{\text{ens}} = \boldsymbol{\theta}\right)$ for any $\boldsymbol{\theta} \in \boldsymbol{\Theta}$. Based on this prior model, we can learn the parametric expert prior using maximum marginal likelihood estimation, as formulated below.

Proposition 3 (Ensemble Marginal Likelihood)**.** *Consider a contextual MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, R, H, \rho, \mu^\star)$. Assume the transition function $\mathcal{T}$ does not depend on the task variables and Assumption 1 holds. Then, the negative marginal log-likelihood of expert data $\mathcal{D}_E$ under the ensemble prior $\mu_{\boldsymbol{\theta}_{\text{ens}}}$ is upper bounded by*

$$-\log \mathrm{P}_E\left(\mathcal{D}_E\,;\,\mu_{\boldsymbol{\theta}_{\text{ens}}}\right) \leq \frac{1}{L}\sum_{i=1}^{L}\sum_{\tau\in\mathcal{D}_E}\sum_{(s,a)\in\tau} \log\left(\sum_{a'\in\mathcal{A}} \exp\left\{\beta\cdot Q\left(s, a'\,;\,\boldsymbol{\theta}^i_{\text{ens}}\right)\right\}\right) - \beta\cdot Q\left(s, a\,;\,\boldsymbol{\theta}^i_{\text{ens}}\right),$$

where β is the competence level of the expert in Assumption 1.

Proposition 3 is proved in Appendix A.6. We can then initialize the Q-networks in the Bootstrapped DQN method using ensemble parameters that minimize the above upper bound. We will refer to this method as `ExPerior-Param`. As an alternative approach, instead of minimizing the above upper bound, we can match the discrete prior distribution $\mu_{\boldsymbol{\theta}_{\text{ens}}}$ to the max-entropy prior by initializing the Q-functions in the ensemble with parameters sampled from the max-entropy expert prior. In particular, we can apply SGLD on the log-pdf of the max-entropy prior derived in Appendix A.5. We will refer to this approach as `ExPerior-MaxEnt`.

Experimental Setup. A main challenge in RL is the reward *sparsity*, where the learner needs to explore the environment deeply to observe reward states. Utilizing expert demonstrations can significantly improve the efficiency of exploration. For this reason, we focus on "Deep Sea," a sparse-reward tabular RL environment proposed by Osband et al. [55] to assess deep exploration for different RL methods. The environment is an $M \times M$ grid, where the agent starts at the top-left corner of the map, and at each time step, it chooses an action from $\mathcal{A} = \{\texttt{left}, \texttt{right}\}$ to move to the left or right column, while going down by one row. In the original version of Deep Sea, the goal is always on the bottom-right corner of the map. We introduce unobserved task variables by defining a distribution over the goal columns while keeping the goal row the same. We consider four types of goal distributions where the goal is situated at (1) the bottom-right corner of the grid, (2) uniformly at the bottom of any of the right-most $\frac{M}{4}$ columns, (3) uniformly at the bottom of any of the right-most $\frac{M}{2}$ columns, and (4) uniformly at the bottom of any of the M columns. We set $M = 30$ and generate $N = 1000$ samples from the optimal policies as offline expert demonstrations. To further evaluate ExPerior and showcase its applicability to partially-observed MDP, we also consider the "Frozen

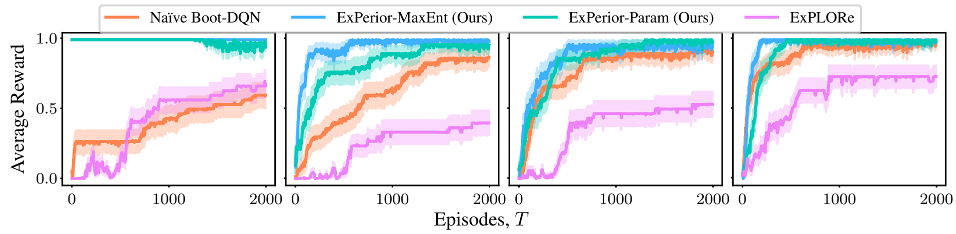

Figure 4: The average reward per episode over 2,000 episodes in "Deep Sea." The goal is located at the right column, uniformly at the right-most quarter of the columns, uniformly at the right-most half, and uniformly at random over all the columns, respectively. ExPerior outperforms the baselines in all instances.

Lake" environment, which requires the learner to navigate to a goal while avoiding hazards [17]. The learner cannot observe the hazard location, while the expert has access to the whole map. Taking action, reaching the goal, and hitting the hazard incur rewards of -2, 20, and -100, respectively. The frozen lake map is 5×5, where the hazard (weak ice) is randomly located in the interior squares. We consider different settings with 2, 5, 7, and 9 potential locations for the hazard. At the start of each episode, the hazard will be chosen randomly within the potential locations. We generate $N = 1000$ samples from noisily rational experts with different competence levels for this environment. See Appendix C.2 for the MDP experiments in Frozen Lake experiments.

Baselines. We compare ExPerior to the following baselines. (1) `ExPLORe`, proposed by Li et al. [38] to accelerate off-policy reinforcement learning using unlabeled prior data. In this method, the offline demonstrations are assigned optimistic reward labels generated using the online data with regular updates. This information is then combined with the buffer data to perform off-policy learning. (2) `Naïve Boot-DQN`, which is the original implementation of Bootstrapped DQN with randomly initialized Q-networks [53]. The latter baseline is purely online.

Deep Sea Results. Figure 4 demonstrates the average reward per episode achieved by the baselines for $T = 2000$ episodes. For each goal distribution, we run the baselines with 30 different seeds and take the average to estimate the expected reward. ExPerior outperforms the baselines in all instances. However, the gap between ExPerior and the fully online `Naïve Boot-DQN`, which measures the effect of using the expert data, decreases as we go from the low-entropy setting (upper left) to the high-entropy task distribution (bottom right). This is consistent with the empirical and theoretical results discussed in section 5 and confirms our expectation that the expert demonstrations may not be helpful under strong unobserved confounding (strong task heterogeneity). The `ExPLORe` baseline substantially underperforms, even compared to the fully online `Naïve Boot-DQN` (except for the first task distribution with zero-entropy). We suspect this is because `ExPLORe` uses actor-critic methods as its backbone model, which are shown to struggle with deep exploration [57].

Frozen Lake Results. We run all the baselines for 90,000 steps with 30 different seeds. Table 1 shows the average reward after 500 evaluation steps at the end of the training. ExPerior outperforms the baselines in almost all instances except for the case of $\beta = 0.1$, which corresponds to a nearly random expert. On the other hand, `ExPLORe` achieves near-optimal results for $\beta = 0.1$. We hypothesize that `ExPLORe`'s performance is mainly due to the superiority of their base actor-critic model since it can achieve near-optimal performance even when the expert trajectories are low-quality.

7 Conclusion

We introduce the Experts-as-Priors (ExPerior) framework, a novel empirical Bayes approach, to address the problem of sequential decision-making using expert demonstrations with unobserved heterogeneity. We ground our methodology in the maximum entropy principle to infer a prior distribution from expert data that guides the learning process in both bandit settings and Markov Decision Processes (MDPs). This advantage underscores the utility of our approach in contexts where the learner faces uncertainty and variability in task parameters, a common challenge in real-world applications from autonomous driving to personalized learning environments. Our work contributes to the understanding of leveraging expert demonstrations under unobserved heterogeneity and offers a practical framework readily applied to a broad spectrum of decision-making tasks. We provide a principled way to incorporate the wealth of information contained in expert behaviours, thus opening new avenues for research in meta-reinforcement learning. One limitation of our work is the limited set of experiments, especially those with human-in-the-loop. Future directions include extending to more complex environments, and further investigating our RL algorithm's theoretical properties.

References

[1] Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Alex Graves, Ioannis Antonoglou, Daan Wierstra, and Martin Riedmiller. Playing atari with deep reinforcement learning. *arXiv preprint arXiv:1312.5602*, 2013.

[2] David Silver, Aja Huang, Chris J Maddison, Arthur Guez, Laurent Sifre, George Van Den Driessche, Julian Schrittwieser, Ioannis Antonoglou, Veda Panneershelvam, Marc Lanctot, et al. Mastering the game of go with deep neural networks and tree search. *nature*, 529 (7587):484–489, 2016.

[3] David Silver, Thomas Hubert, Julian Schrittwieser, Ioannis Antonoglou, Matthew Lai, Arthur Guez, Marc Lanctot, Laurent Sifre, Dharshan Kumaran, Thore Graepel, et al. Mastering chess and shogi by self-play with a general reinforcement learning algorithm. *arXiv preprint arXiv:1712.01815*, 2017.

[4] David Silver, Guy Lever, Nicolas Heess, Thomas Degris, Daan Wierstra, and Martin Riedmiller. Deterministic policy gradient algorithms. In *International conference on machine learning*, pages 387–395. Pmlr, 2014.

[5] Timothy P Lillicrap, Jonathan J Hunt, Alexander Pritzel, Nicolas Heess, Tom Erez, Yuval Tassa, David Silver, and Daan Wierstra. Continuous control with deep reinforcement learning. *arXiv preprint arXiv:1509.02971*, 2015.

[6] Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744, 2022.

[7] Andrew Wagenmaker and Aldo Pacchiano. Leveraging offline data in online reinforcement learning. In *International Conference on Machine Learning*, pages 35300–35338. PMLR, 2023.

[8] Yuda Song, Yifei Zhou, Ayush Sekhari, J Andrew Bagnell, Akshay Krishnamurthy, and Wen Sun. Hybrid rl: Using both offline and online data can make rl efficient. *arXiv preprint arXiv:2210.06718*, 2022.

[9] Philip J Ball, Laura Smith, Ilya Kostrikov, and Sergey Levine. Efficient online reinforcement learning with offline data. In *International Conference on Machine Learning*, pages 1577–1594. PMLR, 2023.

[10] Ashvin Nair, Bob McGrew, Marcin Andrychowicz, Wojciech Zaremba, and Pieter Abbeel. Overcoming exploration in reinforcement learning with demonstrations. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 6292–6299. IEEE, 2018.

[11] Serkan Cabi, Sergio Gómez Colmenarejo, Alexander Novikov, Ksenia Konyushkova, Scott Reed, Rae Jeong, Konrad Zolna, Yusuf Aytar, David Budden, Mel Vecerik, et al. Scaling data-driven robotics with reward sketching and batch reinforcement learning. *arXiv preprint arXiv:1909.12200*, 2019.

[12] Richard A. Briesch, Pradeep K. Chintagunta, and Rosa L. Matzkin. Nonparametric Discrete Choice Models With Unobserved Heterogeneity. *Journal of Business & Economic Statistics*, 28(2):291–307, 2010. ISSN 0735-0015. Publisher: [American Statistical Association, Taylor & Francis, Ltd.].

[13] Lantao Yu, Tianhe Yu, Chelsea Finn, and Stefano Ermon. Meta-inverse reinforcement learning with probabilistic context variables. *Advances in neural information processing systems*, 32, 2019.

[14] Nathan Kallus and Angela Zhou. Minimax-Optimal Policy Learning Under Unobserved Confounding. *Management Science*, 67(5):2870–2890, May 2021. ISSN 0025-1909, 1526-5501. doi: 10.1287/mnsc.2020.3699.

[15] Andrew Bennett, Nathan Kallus, Lihong Li, and Ali Mousavi. Off-policy Evaluation in Infinite-Horizon Reinforcement Learning with Latent Confounders. In *Proceedings of The 24th International Conference on Artificial Intelligence and Statistics*, pages 1999–2007. PMLR, March 2021. ISSN: 2640-3498.

[16] Sanjiban Choudhury, Mohak Bhardwaj, Sankalp Arora, Ashish Kapoor, Gireeja Ranade, Sebastian Scherer, and Debadeepta Dey. Data-driven planning via imitation learning. *The International Journal of Robotics Research*, 37(13-14):1632–1672, 2018.

[17] Andrew Warrington, Jonathan W Lavington, Adam Scibior, Mark Schmidt, and Frank Wood. Robust asymmetric learning in pomdps. In *International Conference on Machine Learning*, pages 11013–11023. PMLR, 2021.

[18] Aaron Walsman, Muru Zhang, Sanjiban Choudhury, Dieter Fox, and Ali Farhadi. Impossibly good experts and how to follow them. In *The Eleventh International Conference on Learning Representations*, 2022.

[19] Idan Shenfeld, Zhang-Wei Hong, Aviv Tamar, and Pulkit Agrawal. TGRL: An algorithm for teacher guided reinforcement learning. In Andreas Krause, Emma Brunskill, Kyunghyun Cho, Barbara Engelhardt, Sivan Sabato, and Jonathan Scarlett, editors, *Proceedings of the 40th International Conference on Machine Learning*, volume 202 of *Proceedings of Machine Learning Research*, pages 31077–31093. PMLR, 23–29 Jul 2023.

[20] Junzhe Zhang, Daniel Kumor, and Elias Bareinboim. Causal imitation learning with unobserved confounders. *Advances in neural information processing systems*, 33:12263–12274, 2020.

[21] Gokul Swamy, Sanjiban Choudhury, J Bagnell, and Steven Z Wu. Sequence model imitation learning with unobserved contexts. *Advances in Neural Information Processing Systems*, 35: 17665–17676, 2022.

[22] Botao Hao, Rahul Jain, Tor Lattimore, Benjamin Van Roy, and Zheng Wen. Leveraging demonstrations to improve online learning: Quality matters. In *International Conference on Machine Learning*, pages 12527–12545. PMLR, 2023.

[23] Botao Hao, Rahul Jain, Dengwang Tang, and Zheng Wen. Bridging imitation and online reinforcement learning: An optimistic tale. *arXiv preprint arXiv:2303.11369*, 2023.

[24] Luca Weihs, Unnat Jain, Iou-Jen Liu, Jordi Salvador, Svetlana Lazebnik, Aniruddha Kembhavi, and Alex Schwing. Bridging the imitation gap by adaptive insubordination. *Advances in Neural Information Processing Systems*, 34:19134–19146, 2021.

[25] Leonardo Cella, Alessandro Lazaric, and Massimiliano Pontil. Meta-learning with stochastic linear bandits. In *International Conference on Machine Learning*, pages 1360–1370. PMLR, 2020.

[26] Leonardo Cella and Massimiliano Pontil. Multi-task and meta-learning with sparse linear bandits. In *Uncertainty in Artificial Intelligence*, pages 1692–1702. PMLR, 2021.

[27] Ian Osband, Daniel Russo, and Benjamin Van Roy. (more) efficient reinforcement learning via posterior sampling. *Advances in Neural Information Processing Systems*, 26, 2013.

[28] Aravind Rajeswaran, Vikash Kumar, Abhishek Gupta, Giulia Vezzani, John Schulman, Emanuel Todorov, and Sergey Levine. Learning complex dexterous manipulation with deep reinforcement learning and demonstrations. *arXiv preprint arXiv:1709.10087*, 2017.

[29] Ashvin Nair, Abhishek Gupta, Murtaza Dalal, and Sergey Levine. Awac: Accelerating online reinforcement learning with offline datasets. *arXiv preprint arXiv:2006.09359*, 2020.

[30] Mel Vecerik, Todd Hester, Jonathan Scholz, Fumin Wang, Olivier Pietquin, Bilal Piot, Nicolas Heess, Thomas Rothörl, Thomas Lampe, and Martin Riedmiller. Leveraging demonstrations for deep reinforcement learning on robotics problems with sparse rewards. *arXiv preprint arXiv:1707.08817*, 2017.

[31] Todd Hester, Matej Vecerik, Olivier Pietquin, Marc Lanctot, Tom Schaul, Bilal Piot, Dan Horgan, John Quan, Andrew Sendonaris, Ian Osband, et al. Deep q-learning from demonstrations. In *Proceedings of the AAAI conference on artificial intelligence*, 2018.

[32] Abhishek Gupta, Russell Mendonca, YuXuan Liu, Pieter Abbeel, and Sergey Levine. Meta-reinforcement learning of structured exploration strategies. *Advances in neural information processing systems*, 31, 2018.

[33] Anusha Nagabandi, Ignasi Clavera, Simin Liu, Ronald S Fearing, Pieter Abbeel, Sergey Levine, and Chelsea Finn. Learning to adapt in dynamic, real-world environments through meta-reinforcement learning. *arXiv preprint arXiv:1803.11347*, 2018.

[34] Jacob Beck, Risto Vuorio, Evan Zheran Liu, Zheng Xiong, Luisa Zintgraf, Chelsea Finn, and Shimon Whiteson. A survey of meta-reinforcement learning. *arXiv preprint arXiv:2301.08028*, 2023.

[35] Jay Mardia, Jiantao Jiao, Ervin Tánczos, Robert D Nowak, and Tsachy Weissman. Concentration inequalities for the empirical distribution of discrete distributions: beyond the method of types. *Information and Inference: A Journal of the IMA*, 9(4):813–850, 2020.

[36] Vinay Kumar Verma, Dhanajit Brahma, and Piyush Rai. Meta-learning for generalized zero-shot learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 6062–6069, 2020.

[37] Julie Jiang, Kristina Lerman, and Emilio Ferrara. Zero-shot meta-learning for small-scale data from human subjects. In *2023 IEEE 11th International Conference on Healthcare Informatics (ICHI)*, pages 311–320. IEEE, 2023.

[38] Qiyang Li, Jason Zhang, Dibya Ghosh, Amy Zhang, and Sergey Levine. Accelerating exploration with unlabeled prior data. *Advances in Neural Information Processing Systems*, 36, 2024.

[39] Russell Mendonca, Abhishek Gupta, Rosen Kralev, Pieter Abbeel, Sergey Levine, and Chelsea Finn. Guided meta-policy search. *Advances in Neural Information Processing Systems*, 32, 2019.

[40] Allan Zhou, Eric Jang, Daniel Kappler, Alex Herzog, Mohi Khansari, Paul Wohlhart, Yunfei Bai, Mrinal Kalakrishnan, Sergey Levine, and Chelsea Finn. Watch, try, learn: Meta-learning from demonstrations and reward. *arXiv preprint arXiv:1906.03352*, 2019.

[41] Kate Rakelly, Aurick Zhou, Chelsea Finn, Sergey Levine, and Deirdre Quillen. Efficient off-policy meta-reinforcement learning via probabilistic context variables. In *International conference on machine learning*, pages 5331–5340. PMLR, 2019.

[42] Daniel J Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, Zheng Wen, et al. A tutorial on thompson sampling. *Foundations and Trends® in Machine Learning*, 11(1):1–96, 2018.

[43] Assaf Hallak, Dotan Di Castro, and Shie Mannor. Contextual markov decision processes. *arXiv preprint arXiv:1502.02259*, 2015.

[44] Bradley P Carlin and Thomas A Louis. Empirical bayes: Past, present and future. *Journal of the American Statistical Association*, 95(452):1286–1289, 2000.

[45] Michael C Fu. Chapter 19 gradient estimation. *Simulation*, 13:575–616, 2006.

[46] R Tyrrell Rockafellar. *Convex analysis*, volume 11. Princeton university press, 1997.

[47] Miroslav Dudík, Steven J Phillips, and Robert E Schapire. Maximum entropy density estimation with generalized regularization and an application to species distribution modeling. *Journal of Machine Learning Research*, 2007.

[48] Max Welling and Yee W Teh. Bayesian learning via stochastic gradient langevin dynamics. In *Proceedings of the 28th international conference on machine learning (ICML-11)*, pages 681–688, 2011.

[49] Daniel Russo and Benjamin Van Roy. An information-theoretic analysis of thompson sampling. *The Journal of Machine Learning Research*, 17(1):2442–2471, 2016.

[50] Eyal Even-Dar, Shie Mannor, and Yishay Mansour. Pac bounds for multi-armed bandit and markov decision processes. In *Computational Learning Theory: 15th Annual Conference on Computational Learning Theory, COLT 2002 Sydney, Australia, July 8–10, 2002 Proceedings 15*, pages 255–270. Springer, 2002.

[51] Vikranth Dwaracherla and Benjamin Van Roy. Langevin dqn. *arXiv preprint arXiv:2002.07282*, 2020.

[52] Haque Ishfaq, Qingfeng Lan, Pan Xu, A. Rupam Mahmood, Doina Precup, Anima Anandkumar, and Kamyar Azizzadenesheli. Provable and practical: Efficient exploration in reinforcement learning via langevin monte carlo. In *The Twelfth International Conference on Learning Representations*, 2024.

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

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

[55] Ian Osband, Benjamin Van Roy, Daniel J Russo, Zheng Wen, et al. Deep exploration via randomized value functions. *J. Mach. Learn. Res.*, 20(124):1–62, 2019.

[56] Ian Osband, Zheng Wen, Seyed Mohammad Asghari, Vikranth Dwaracherla, Morteza Ibrahimi, Xiuyuan Lu, and Benjamin Van Roy. Approximate thompson sampling via epistemic neural networks. In *Uncertainty in Artificial Intelligence*, pages 1586–1595. PMLR, 2023.

[57] Ian Osband, Yotam Doron, Matteo Hessel, John Aslanides, Eren Sezener, Andre Saraiva, Katrina McKinney, Tor Lattimore, Csaba Szepesvari, Satinder Singh, et al. Behaviour suite for reinforcement learning. *arXiv preprint arXiv:1908.03568*, 2019.

[58] Jonathan M Borwein and Qiji J Zhu. Techniques of variational analysis, 2004.

A Proofs

A.1 Notation

We assume $\mathcal{C}$ is a measurable set with an appropriate σ-algebra and there exists a probability measure μ_0 on $\mathcal{C}$. We denote $L^p(\mathcal{C}, \mu_0)$ as the space of all measurable functions $f : \mathcal{C} \to \mathbb{R}$ such that $\|f\|_p = \left(\int_{\mathcal{C}} |f|^p \, \mathrm{d}\mu_0\right)^{1/p} < \infty$. Moreover, we define $L^\infty(\mathcal{C}, \mu_0)$ as the space of all essentially bounded measurable functions from $\mathcal{C}$ to $\mathbb{R}$. Unless stated otherwise, we assume the probability measures are absolutely continuous w.r.t. μ_0, and their density functions are in $L^1(\mathcal{C}, \mu_0)$. We may abuse the notation and use the same symbol for a probability measure and its Radon–Nikodym derivative w.r.t. μ_0. Finally, we use $\mathbb{E}[\cdot]$ to denote expectation under the probability measure μ_0.

A.2 Useful Lemmas

Here, we state and prove a set of results that will be useful for the rest of this section. The first one is Fenchel's duality theorem:

Lemma 4 (Fenchel's Duality [58])**.** *Let X and Y be Banach spaces, let $f : X \to \mathbb{R} \cup \{+\infty\}$ and $g : Y \to \mathbb{R} \cup \{+\infty\}$ be convex functions and let $A : X \to Y$ be a bounded linear map. Define the primal and dual values $p, d \in [-\infty, +\infty]$ by the Fenchel problems*

$$
\begin{aligned}
p &= \inf_{x \in X} f(x) + g(Ax) \\
d &= \sup_{y^* \in Y^*} -f^*(A^* y^*) - g^*(-y^*),
\end{aligned}
$$

where f^* and g^* are the Fenchel conjugates of f and g defined as $f^*(x^*) = \sup_{x \in X} \langle x^*, x \rangle - f(x)$ (similarly for g), X^* is the dual space of X and $\langle \cdot, \cdot \rangle$ is its duality pairing, and $A^* : Y^\star \to X^\star$ is the adjoint operator of A, i.e., $\langle A^* y^*, x \rangle = \langle y^*, Ax \rangle$. Suppose A dom$(f) \cap$ cont$(g) \neq \emptyset$, where dom$(f) := \{x \in X \,;\, f(x) < \infty\}$ and cont(g) are the continuous points of g. Then, strong duality holds, i.e., $p = d$.

Proof. See the proof of Theorem 4.4.3 in Borwein and Zhu [58]. $\square$

We can use Fenchel's duality to solve generalized maximum entropy problems. In particular, we prove a generalization of Theorem 2 in [47] for density functions in $L^1(\mathcal{C}, \mu_0)$:

Lemma 5. *For any function $\mu \in L^1(\mathcal{C}, \mu_0)$, define the extended KL divergence as*

$$
\psi(\mu) := \begin{cases} \mathrm{D_{KL}}\left(\mu \,\|\, \mu_0\right) & \text{If } \|\mu\|_1 = 1, \\ +\infty & \text{o.w.} \end{cases}
$$

Moreover, assume a set of bounded feature functions $m_1, m_2, \ldots, m_N : \mathcal{C} \to \mathbb{R}$ is given and denote $\mathbf{m}$ as the vector of all N features. Consider the linear function $A_{\mathbf{m}} : L^1(\mathcal{C}, \mu_0) \to \mathbb{R}^N$ defined as

$$
\forall \mu \in L^1(\mathcal{C}, \mu_0): \;\; A_{\mathbf{m}}(\mu) := \left(\mathbb{E}\left[m_1 \cdot \mu\right], \mathbb{E}\left[m_2 \cdot \mu\right], \ldots, \mathbb{E}\left[m_N \cdot \mu\right]\right).
$$

We define the generalized maximum entropy problem as the following:

$$
\inf_{\mu \in L^1(\mathcal{C}, \mu_0)} \psi(\mu) + \zeta\left(A_{\mathbf{m}}(\mu)\right), \tag{5}
$$

for an arbitrary closed proper convex function $\zeta : \mathbb{R}^N \to \mathbb{R}$. Then the following holds:

1. *The dual optimization of (5) is given by*

$$
\sup_{\boldsymbol{\alpha} \in \mathbb{R}^N} -\log \mathbb{E}\left[\exp\left\{\mathbf{m}^\top \boldsymbol{\alpha}\right\}\right] - \zeta^*\left(-\boldsymbol{\alpha}\right), \tag{6}
$$

   where $\zeta^\star$ is the convex conjugate function of ζ.

2. *Denote $\boldsymbol{\alpha}^1, \boldsymbol{\alpha}^2, \ldots$ as a sequence in $\mathbb{R}^N$ converging to supremum (6), and define the following Gibbs density functions*

$$
\mu^{\boldsymbol{\alpha}}_{\mathit{Gibbs}}(c) := \frac{\exp\left\{\mathbf{m}(c)^\top \boldsymbol{\alpha}\right\}}{\mathbb{E}\left[\exp\left\{\mathbf{m}^\top \boldsymbol{\alpha}\right\}\right]}.
$$

   Then,

$$
\inf_{\mu \in L^1(\mathcal{C}, \mu_0)} \psi(\mu) + \zeta\left(A_{\mathbf{m}}(\mu)\right) = \lim_{n \to \infty} \psi(\mu^{\boldsymbol{\alpha}^n}_{\mathit{Gibbs}}) + \zeta\left(A_{\mathbf{m}}(\mu^{\boldsymbol{\alpha}^n}_{\mathit{Gibbs}})\right).
$$

Proof. **Part 1:** We first derive the convex conjugate of ψ. Note that $\left(L^1(\mathcal{C},\mu_0)\right)^\star = L^\infty(\mathcal{C},\mu_0)$ with the pairing

$$\forall h \in L^\infty(\mathcal{C},\mu_0),\ \mu \in L^1(\mathcal{C},\mu_0):\ \langle h,\mu\rangle := \int_{\mathcal{C}} h(c)\cdot\mu(c)\,\mathrm{d}\mu_0.$$

Hence, by Donsker and Varadhan's variational formula

$$\forall h \in L^\infty(\mathcal{C},\mu_0):\ \psi^\star(h) = \sup_{\mu\in L^1(\mathcal{C},\mu_0)} \langle h,\mu\rangle - \psi(\mu) = \log\mathbb{E}\left[\exp\{h\}\right]. \tag{7}$$

Moreover, the adjoint operator of $A_{\mathbf{m}}$ is given by $A^\star_{\mathbf{m}}: \mathbb{R}^N \to (\mathcal{C}\to\mathbb{R})$:

$$\forall \boldsymbol{\alpha}\in\mathbb{R}^N,\ c\in\mathcal{C}:\ A^\star_{\mathbf{m}}(\boldsymbol{\alpha})(c) = \mathbf{m}(c)^\top\boldsymbol{\alpha}. \tag{8}$$

Using (7) and (8) and Lemma 4 concludes the proof.

Part 2: Denote the primal and dual objective functions by

$$\begin{aligned}P(\mu) &:= \psi(\mu) + \zeta\left(A_{\mathbf{m}}(\mu)\right),\\ D(\boldsymbol{\alpha}) &:= -\log\mathbb{E}\left[\exp\left\{\mathbf{m}^\top\boldsymbol{\alpha}\right\}\right] - \zeta^*\left(-\boldsymbol{\alpha}\right),\end{aligned}$$

and their optimal values as P^* and D^*. For any $\nu\in L^1(\mathcal{C},\mu_0)$, note that

$$\begin{aligned}\mathrm{D}_{\mathrm{KL}}\left(\nu \,\|\, \mu_0\right) - \mathrm{D}_{\mathrm{KL}}\left(\nu \,\|\, \mu^{\boldsymbol{\alpha}}_{\text{Gibbs}}\right) &= \int_{\mathcal{C}} \nu\log\nu\,\mathrm{d}\mu_0 - \left(\int_{\mathcal{C}}\nu\log\nu\,\mathrm{d}\mu_0 - \int_{\mathcal{C}}\nu\log\mu^{\boldsymbol{\alpha}}_{\text{Gibbs}}\,\mathrm{d}\mu_0\right)\\ &= \int_{\mathcal{C}}\left(\mathbf{m}(c)^\top\boldsymbol{\alpha}\right)\nu(c)\,\mathrm{d}\mu_0 - \log\mathbb{E}\left[\exp\left\{\mathbf{m}^\top\boldsymbol{\alpha}\right\}\right]\\ &= A_{\mathbf{m}}(\nu)^\top\boldsymbol{\alpha} - \log\mathbb{E}\left[\exp\left\{\mathbf{m}^\top\boldsymbol{\alpha}\right\}\right].\end{aligned} \tag{9}$$

Using (9), we can re-write the dual objective function as:

$$\forall\boldsymbol{\alpha}\in\mathbb{R}^N, \nu\in L^1(\mathcal{C},\mu_0):\quad D(\boldsymbol{\alpha}) = -\mathrm{D}_{\mathrm{KL}}\left(\nu\,\|\,\mu^{\boldsymbol{\alpha}}_{\text{Gibbs}}\right) + \mathrm{D}_{\mathrm{KL}}\left(\nu\,\|\,\mu_0\right) - A_{\mathbf{m}}(\nu)^\top\boldsymbol{\alpha} - \zeta^\star(-\boldsymbol{\alpha}). \tag{10}$$

Moreover, note that

$$\begin{aligned}-A_{\mathbf{m}}(\nu)^\top\boldsymbol{\alpha} - \zeta^\star(-\boldsymbol{\alpha}) &= -A_{\mathbf{m}}(\nu)^\top\boldsymbol{\alpha} - \left(\sup_x \langle x, -\alpha\rangle - \zeta(x)\right)\\ &\le -A_{\mathbf{m}}(\nu)^\top\boldsymbol{\alpha} - \Big(\langle A_{\mathbf{m}}(\nu), -\alpha\rangle - \zeta(A_{\mathbf{m}}(\nu))\Big)\\ &= \zeta(A_{\mathbf{m}}(\nu)).\end{aligned} \tag{11}$$

Combining (10) and (11), we get

$$\begin{aligned}\forall\boldsymbol{\alpha}\in\mathbb{R}^N, \nu\in L^1(\mathcal{C},\mu_0):\quad D(\boldsymbol{\alpha}) &\le -\mathrm{D}_{\mathrm{KL}}\left(\nu\,\|\,\mu^{\boldsymbol{\alpha}}_{\text{Gibbs}}\right) + \mathrm{D}_{\mathrm{KL}}\left(\nu\,\|\,\mu_0\right) + \zeta(A_{\mathbf{m}}(\nu))\\ &= -\mathrm{D}_{\mathrm{KL}}\left(\nu\,\|\,\mu^{\boldsymbol{\alpha}}_{\text{Gibbs}}\right) + P(\nu).\end{aligned} \tag{12}$$

Now, fix an arbitrary $\epsilon > 0$, and consider a sequence of $\mu^1, \mu^2, \ldots \in L^1(\mathcal{C},\mu_0)$ such that for all $j\in\mathbb{N}$:

$$P(\mu^j) - P^* < \frac{\epsilon}{2^j}. \tag{13}$$

We can re-write (13) using the fact $P^* = D^* = \lim_{n\to\infty} D(\boldsymbol{\alpha}^n)$:

$$\forall j\in\mathbb{N}:\quad \lim_{n\to\infty} P(\mu^j) - D(\boldsymbol{\alpha}^n) < \frac{\epsilon}{2^j} \tag{14}$$

In particular, by setting $\nu = \mu^j$ in (12) and combining the result with (14), we get

$$\forall j\in\mathbb{N}:\quad \lim_{n\to\infty}\mathrm{D}_{\mathrm{KL}}\left(\mu^j \,\middle\|\, \mu^{\boldsymbol{\alpha}^n}_{\text{Gibbs}}\right) < \frac{\epsilon}{2^j}.$$

Hence, $\lim_{j\in\infty}\lim_{n\to\infty}\mathrm{D}_{\mathrm{KL}}\left(\mu^j \,\middle\|\, \mu^{\boldsymbol{\alpha}^n}_{\text{Gibbs}}\right) = 0$. From properties of the KL divergence, it follows that $\lim_{j\to\infty} P(\mu^j) = \lim_{n\to\infty} P(\mu^{\boldsymbol{\alpha}^n}_{\text{Gibbs}})$, concluding the proof. $\square$

A.3 Max-Entropy Prior

Proposition 1. *Let $N = |\mathcal{D}_E|$ be the number of demonstrations in $\mathcal{D}_E$. For each $c \in \mathcal{C}$ and demonstration $\tau_E = (s_1, a_1, s_2, a_2, \ldots, s_H, a_H, s_{H+1}) \in \mathcal{D}_E$, define $m_{\tau_E}(c)$ as the (partial) likelihood of τ_E under c:*

$$m_{\tau_E}(c) = \prod_{h=1}^{H} p_E\left(a_h \mid s_h\,;\, c\right) \mathcal{T}\left(s_{h+1} \mid s_h, a_h, c\right). \tag{15}$$

Denote $\mathbf{m}(c) \in \mathbb{R}^N$ as the vector with elements $m_{\tau_E}(c)$ for $\tau_E \in \mathcal{D}_E$. Moreover, let $\lambda^\star \in \mathbb{R}^{\geq 0}$ be the optimal solution to the Lagrange dual problem of (2). Then, the solution to optimization (2) is as follows:

$$\mu_{ME}(c) = \lim_{n\to\infty} \frac{\exp\left\{\mathbf{m}(c)^\top \boldsymbol{\alpha}_n\right\}}{\mathbb{E}_{c\sim\mu_0}\left[\exp\left\{\mathbf{m}(c)^\top \boldsymbol{\alpha}_n\right\}\right]},$$

where $\{\boldsymbol{\alpha}_n\}_{n=1}^{\infty}$ is a sequence converging to the following supremum:

$$\sup_{\boldsymbol{\alpha}\in\mathbb{R}^N} -\log \mathbb{E}_{c\sim\mu_0}\left[\exp\left\{\mathbf{m}(c)^\top \boldsymbol{\alpha}\right\}\right] + \frac{\lambda^\star}{N}\sum_{i=1}^{N} \log\left(\frac{N\cdot\alpha_i}{\lambda^\star}\right).$$

Proof. We first simplify the KL-divergence between the empirical distribution of the expert trajectories $\widehat{\mathrm{P}}_{\mathrm{E}}$ and the marginal likelihood $\mathrm{P}_{\mathrm{E}}(\cdot\,;\,\mu)$:

$$\begin{aligned}
\mathrm{D}_{\mathrm{KL}}\left(\widehat{\mathrm{P}}_{\mathrm{E}} \,\middle\|\, \mathrm{P}_{\mathrm{E}}(\cdot\,;\,\mu)\right) &= \sum_{\tau^{(i)}\in\mathcal{D}_{\mathrm{E}}} \widehat{\mathrm{P}}_{\mathrm{E}}(\tau^{(i)}) \log \frac{\widehat{\mathrm{P}}_{\mathrm{E}}(\tau^{(i)})}{\mathrm{P}_{\mathrm{E}}\left(\tau^{(i)}\,;\,\mu\right)} \\
&= -\log N - \frac{1}{N}\sum_{\tau^{(i)}\in\mathcal{D}_{\mathrm{E}}} \log \mathrm{P}_{\mathrm{E}}\left(\tau^{(i)}\,;\,\mu\right) && (\widehat{\mathrm{P}}_{\mathrm{E}}(\tau^{(i)}) = \tfrac{1}{N}) \\
&= -\log N - \frac{1}{N}\sum_{\tau^{(i)}\in\mathcal{D}_{\mathrm{E}}} \log \mathbb{E}\left[m_{\tau^{(i)}}\cdot\mu\right] - \frac{1}{N}\sum_{s_1^{(i)}\in\mathcal{D}_{\mathrm{E}}} \log\rho\left(s_1^{(i)}\right). && \text{By (1) and (15)}
\end{aligned}$$

Using the above equality, we can re-write the definition of uncertainty set $\mathcal{P}(\epsilon)$ as

$$\mathcal{P}(\epsilon) = \left\{\mu\,;\, -\frac{1}{N}\sum_{\tau\in\mathcal{D}_{\mathrm{E}}} \log \mathbb{E}\left[m_\tau\cdot\mu\right] - \epsilon - \log N - \frac{1}{N}\sum_{s_1\in\mathcal{D}_{\mathrm{E}}} \log\rho\left(s_1\right) \leq 0\right\}.$$

Therefore, we can re-write the optimization (2) as

$$\mu_{\mathrm{ME}} = \operatorname*{arg\,min}_{\mu\in L^1(\mathcal{C},\mu_0)} \psi(\mu) \quad \text{s.t.} \quad -\frac{1}{N}\sum_{\tau\in\mathcal{D}_{\mathrm{E}}} \log \mathbb{E}\left[m_\tau\cdot\mu\right] - \epsilon - \log N - \frac{1}{N}\sum_{s_1\in\mathcal{D}_{\mathrm{E}}} \log\rho\left(s_1\right) \leq 0, \tag{16}$$

where the extended KL divergence $\psi(\mu)$ is defined as:

$$\psi(\mu) := \begin{cases} \mathrm{D}_{\mathrm{KL}}\left(\mu \,\|\, \mu_0\right) & \text{If } \|\mu\|_1 = 1, \\ +\infty & \text{o.w.} \end{cases}$$

Note that $\mathcal{P}(\epsilon)$ is a convex set. To see this, consider $\mu_1, \mu_2 \in \mathcal{P}(\epsilon)$. Then, for any $0 \leq \lambda \leq 1$, we have $\mu = (1-\lambda)\mu_1 + \lambda\mu_2 \in \mathcal{P}(\epsilon)$ since $\mathbb{E}\left[m_\tau\cdot\mu\right]$ is linear in μ and $-\log$ is convex. Moreover, It is easy to see there exists a strictly feasible solution for (16) (e.g., consider the true distribution $\mu^\star$ over $\mathcal{C}$). Thus, strong duality holds, and we can form the Lagrangian function as

$$L(\mu,\lambda) := \psi(\mu) + \lambda\left(\frac{1}{N}\sum_{\tau\in\mathcal{D}_{\mathrm{E}}} -\log \mathbb{E}\left[m_\tau\cdot\mu\right]\right) - \lambda\left(\epsilon + \log N + \frac{1}{N}\sum_{s_1\in\mathcal{D}_{\mathrm{E}}} \log\rho\left(s_1\right)\right).$$

Given that $\lambda^\star \in \mathbb{R}^{\geq 0}$ is the optimal solution to the Lagrange dual problem, the maximum entropy prior μ_{ME} will be the solution to

$$\inf_{\mu \in L^1(\mathcal{C},\mu_0)} L(\mu,\lambda^\star) = \inf_{\mu \in L^1(\mathcal{C},\mu_0)} \psi(\mu) + \lambda^\star \left(\frac{1}{N} \sum_{\tau \in \mathcal{D}_{\text{E}}} - \log \mathbb{E}\left[m_\tau \cdot \mu\right] \right) + \text{constant in } \mu. \quad (17)$$

Now, for each $\mathbf{x} \in \mathbb{R}^N$, define the convex function $\zeta(\mathbf{x}) := \frac{\lambda^\star}{N} \left(\sum_{i=1}^N - \log x_i \right)$. Moreover, for $\mu \in L^1(\mathcal{C},\mu_0)$, define $A_{\mathbf{m}}(\mu) := \left(\mathbb{E}\left[m_{\tau^{(1)}} \cdot \mu\right], \mathbb{E}\left[m_{\tau^{(2)}} \cdot \mu\right], \dots, \mathbb{E}\left[m_{\tau^{(N)}} \cdot \mu\right] \right)$. Then,

$$L(\mu,\lambda^\star) = \psi(\mu) + \zeta\left(A_{\mathbf{m}}(\mu)\right). \quad (18)$$

Combining (17) and (18), the maximum entropy prior μ_{ME} is the solution to

$$\inf_{\mu \in L^1(\mathcal{C},\mu_0)} \psi(\mu) + \zeta\left(A_{\mathbf{m}}(\mu)\right).$$

Using Lemma 5 and noting that

$$\zeta^*(x^*) = \frac{\lambda^\star}{N} \left(\sum_{i=1}^N -1 - \log\left(-\frac{N}{\lambda^\star} \cdot x_i^* \right) \right)$$

concludes the proof. □

A.4 K-armed Bandit Frequentist Algorithm & Regret

To simplify the analysis, we employ a deterministic sampling approach by pulling each arm a fixed number of times based on its probability. To do so, we discretize the expert policy with a step size $p_{\min}$, which leads to a relative frequency of $\lceil \frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{p_{\min}} \rceil$ for an arm a. In particular, we can choose $p_{\min} = \min_{a \in \mathcal{A}} \widehat{\mathrm{P}}_{\mathrm{E}}(a)$.

Algorithm 1 Successive Elimination with Expert Sampling

Input: Episodes T, Arms $\mathcal{A} = [K]$, expert policy $\widehat{\mathrm{P}}_{\mathrm{E}}$, step size $p_{\min}$, an unknown task $c \sim \mu^\star$, and $\delta \in (0,1)$.
for $t = 1 \dots T$ **do**
  Try an active arm a with a relative frequency of $\lceil \frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{p_{\min}} \rceil$. // all arms are active at $t = 0$.
  // $n_t(a)$ is the number of times that an arm a is pulled by episode t and $\overline{V_c^t}(a)$ is its empirical mean reward.
  Increment $n_t(a)$ and update $\overline{V_c^t}(a)$.
  Construct $\text{UCB}_a^t = \overline{V_c^t}(a) + \sqrt{\frac{\log(4T^4K/\delta)}{2n_t(a)}}$ and $\text{LCB}_a^t = \overline{V_c^t}(a) - \sqrt{\frac{\log(4T^4K/\delta)}{2n_t(a)}}$.
  De-activate all arms a s.t. $\exists a'$ with $\text{UCB}_a \leq \text{LCB}_{a'}$, and normalize $\widehat{\mathrm{P}}_{\mathrm{E}}$.
end for

Theorem 2. *Assume that (i) the mean value of reward function R is bounded in $[0,1]$ for all arms, (ii) $T \cdot p_{\min} \geq 1$, (iii) the expert is optimal, i.e., $\widehat{\mathrm{P}}_E = \mathrm{P}_E(\cdot\,;\mu^\star)$ ($\beta \to \infty, |\mathcal{D}_E| \to \infty$), and (iv) the learner follows Algorithm 1. Then, with probability at least $1-\delta$,*

$$\textit{Reg} \lesssim \sqrt{T \log\left(TK/\delta\right)} \sum_{a,a' \in \mathcal{A}; a \neq a'} \sqrt{\frac{\widehat{\mathrm{P}}_E(a)}{\widehat{\mathrm{P}}_E(a) + \widehat{\mathrm{P}}_E(a')} \cdot \left(1 - \frac{\widehat{\mathrm{P}}_E(a)}{\widehat{\mathrm{P}}_E(a) + \widehat{\mathrm{P}}_E(a')}\right)} \left[\sqrt{\widehat{\mathrm{P}}_E(a)} + \sqrt{\widehat{\mathrm{P}}_E(a')}\right].$$

Proof. Fix $\delta \in (0,1)$ and $c \in \mathcal{C}$. Let $\mathcal{E}$ be the event that $\left|\overline{V_c^t}(a) - V_c(a)\right| \leq \sqrt{\frac{\log(4T^4K/\delta)}{2n_t(a)}}$ for all arms $a \in \mathcal{A}$, all $t \leq T$, and all $T \in \mathbb{N}$, where $n_t(a)$ is the number of times that arm a was pulled by time t. Note that since $T \geq \frac{1}{p_{\min}}$, each arm will be pulled at least once and $n_t(a) \geq 1$.

We first show that $\mathbb{P}(\mathcal{E}) \geq 1-\delta$. Fix T, arm a, and $t \leq T$. Suppose $n_t(a) = j$ for $1 \leq j \leq T$. By Hoeffding's inequality, we have

$$\mathbb{P}\left(\left|\overline{V_c^t}(a) - V_c(a)\right| \leq \sqrt{\frac{\log(4T^4K/\delta)}{2j}}\right) \geq 1 - \frac{\delta}{2T^4K}. \tag{19}$$

Now, using the union bound over all episodes and all actions, we get

$$\begin{aligned}
&\mathbb{P}\left(\exists a \in \mathcal{A}, T \in \mathbb{N}, t \leq T, j \leq t: \ \left|\overline{V_c^t}(a) - V_c(a)\right| > \sqrt{\frac{\log(2T^4K/\delta)}{2j}}\right) \\
&\leq \sum_{T=1}^{\infty}\sum_{a\in\mathcal{A}}\sum_{t=1}^{T}\sum_{j=1}^{t}\mathbb{P}\left(\left|\overline{V_c^t}(a) - V_c(a)\right| > \sqrt{\frac{\log(2T^4K/\delta)}{2j}}\right) \\
&\leq \sum_{T=1}^{\infty}\sum_{a\in\mathcal{A}}\sum_{t=1}^{T} t\cdot\frac{\delta}{2T^4K} && \text{By (19)} \\
&\leq \sum_{T=1}^{\infty}\frac{\delta}{2T^4K}\times T^2\times K = \sum_{T=1}^{\infty}\frac{\delta}{2T^2} \leq \delta,
\end{aligned}$$

which concludes that $\mathbb{P}(\mathcal{E}) \geq 1-\delta$.

The rest of the proof computes the regret for when $\mathcal{E}$ holds. For simplicity and without loss of generality, we assume all expert probabilities are dividable by $p_{\min}$. Recall that we follow a deterministic sampling approach and choose each arm according to its relative frequency $\frac{\widehat{\mathrm{P}}_{\mathrm{E}}(\cdot)}{p_{\min}}$ for multiple batches, where each batch loops over all active actions. Let t_a be the episode in which we eliminate an arm a in favour of another arm. Then, it is easy to show that

$$\forall a' \in \text{active arms by } t_a: \ \widehat{\mathrm{P}}_{\mathrm{E}}(a')\cdot t_a \leq n_{t_a}(a'), \tag{20}$$

This lower bound corresponds to the case where no other arm is eliminated before eliminating a. Moreover, we have an upper bound for $n_{t_a}(a)$ considering the worst-case scenario in which the only remaining arms are a and a_c, where a_c is the optimal action for task c:

$$n_{t_a}(a) \leq \frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{\widehat{\mathrm{P}}_{\mathrm{E}}(a) + \widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}\cdot t_a. \tag{21}$$

Now, let $\mathrm{Reg}_c(a)$ be the total regret contributed by the arm a for a given task $c \sim \mathcal{C}$. We can upper bound the regret as

$$\begin{aligned}
\mathrm{Reg}_c(a) &= n_{t_a}(a)\left(V_c(a_c) - V_c(a)\right) \\
&\overset{(i)}{\leq} 2n_{t_a}(a)\left(\sqrt{\frac{\log(4T^4K/\delta)}{2n_{t_a}(a)}} + \sqrt{\frac{\log(4T^4K/\delta)}{2n_{t_a}(a_c)}}\right) \\
&\leq 2\frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{\widehat{\mathrm{P}}_{\mathrm{E}}(a) + \widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}\cdot t_a\left(\sqrt{\frac{\log(4T^4K/\delta)}{2n_{t_a}(a)}} + \sqrt{\frac{\log(4T^4K/\delta)}{2n_{t_a}(a_c)}}\right) && \text{By (21)} \\
&= \sqrt{2\log(4T^4K/\delta)}\cdot\frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{\widehat{\mathrm{P}}_{\mathrm{E}}(a) + \widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}\cdot t_a\left(\sqrt{\frac{1}{n_{t_a}(a)}} + \sqrt{\frac{1}{n_{t_a}(a_c)}}\right) \\
&\leq \sqrt{2\log(4T^4K/\delta)}\cdot\frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{\widehat{\mathrm{P}}_{\mathrm{E}}(a) + \widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}\cdot t_a\left(\sqrt{\frac{1}{t_a\widehat{\mathrm{P}}_{\mathrm{E}}(a)}} + \sqrt{\frac{1}{t_a\widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}}\right) && \text{By (20)} \\
&= \sqrt{2t_a\log(4T^4K/\delta)}\cdot\frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{\widehat{\mathrm{P}}_{\mathrm{E}}(a) + \widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}\left(\sqrt{\frac{1}{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}} + \sqrt{\frac{1}{\widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}}\right) \\
&\overset{(ii)}{\leq} \sqrt{2T\log(4T^4K/\delta)}\cdot\frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{\widehat{\mathrm{P}}_{\mathrm{E}}(a) + \widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}\left(\sqrt{\frac{1}{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}} + \sqrt{\frac{1}{\widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}}\right),
\end{aligned}$$

where (i) holds since the confidence intervals of arm a and a_c overlap at episode t_a (otherwise, a would have been eliminated before t_a), and (ii) follows from the fact that $t_a \leq T$.

Finally, we upper bound the Bayesian regret by taking the expectation of $\sum_{a \neq a_c} \mathrm{Reg}_c(a)$ over $c \sim \mathcal{C}$. Note that since the expert is optimal, we have $\widehat{\mathrm{P}}_{\mathrm{E}}(a) = \mu^\star(a_c = a)$ for all $k \in \mathcal{A}$.

$$
\begin{aligned}
\mathrm{Reg} &= \mathbb{E}_{c \sim \mu^\star}\left[\sum_{a \neq a_c} \mathrm{Reg}_c(a)\right] \\
&\overset{(i)}{\leq} \sum_{a' \in \mathcal{A}} \mu^\star\left(a_c = a'\right)\left(\max_{c;a_c=a'} \sum_{a \neq a'} \mathrm{Reg}_c(a)\right) \\
&= \sum_{a' \in \mathcal{A}} \widehat{\mathrm{P}}_{\mathrm{E}}(a')\left(\max_{c;a_c=a'} \sum_{a \neq a'} \mathrm{Reg}_c(a)\right) \\
&\leq \sqrt{2T \log\left(4T^4K/\delta\right)} \sum_{a' \in \mathcal{A}} \sum_{a \neq a'} \frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a')\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{\widehat{\mathrm{P}}_{\mathrm{E}}(a) + \widehat{\mathrm{P}}_{\mathrm{E}}(a')}\left(\sqrt{\frac{1}{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}} + \sqrt{\frac{1}{\widehat{\mathrm{P}}_{\mathrm{E}}(a')}}\right) \\
&\overset{(ii)}{\leq} \sqrt{8T \log\left(4TK/\delta\right)} \sum_{a,a' \in \mathcal{A}; a \neq a'} \frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a')\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{\widehat{\mathrm{P}}_{\mathrm{E}}(a) + \widehat{\mathrm{P}}_{\mathrm{E}}(a')}\left(\sqrt{\frac{1}{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}} + \sqrt{\frac{1}{\widehat{\mathrm{P}}_{\mathrm{E}}(a')}}\right) \\
&= \sqrt{8T \log\left(4TK/\delta\right)} \sum_{a,a' \in \mathcal{A}; a \neq a'} \sqrt{\frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a')}{\widehat{\mathrm{P}}_{\mathrm{E}}(a) + \widehat{\mathrm{P}}_{\mathrm{E}}(a')} \cdot \frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{\widehat{\mathrm{P}}_{\mathrm{E}}(a) + \widehat{\mathrm{P}}_{\mathrm{E}}(a')}}\left(\sqrt{\widehat{\mathrm{P}}_{\mathrm{E}}(a)} + \sqrt{\widehat{\mathrm{P}}_{\mathrm{E}}(a')}\right)
\end{aligned}
$$

where (i) follows by partitioning $\mathcal{C}$ into $\{c\,;\, c \in \mathcal{C},\ a_c = a'\}_{a' \in \mathcal{A}}$ and choosing the worst-case task in each partition, and (ii) holds since $4K/\delta > 1$. Replacing $\frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a)}{\widehat{\mathrm{P}}_{\mathrm{E}}(a)+\widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}$ with $1 - \frac{\widehat{\mathrm{P}}_{\mathrm{E}}(a')}{\widehat{\mathrm{P}}_{\mathrm{E}}(a)+\widehat{\mathrm{P}}_{\mathrm{E}}(a_c)}$ concludes the proof. □

A.5 Max-Entropy Expert Posterior for MDPs

Proposition 6 (Max-Entropy Expert Posterior for MDPs)**.** *Consider a contextual MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, R, H, \rho, \mu^\star)$. Assume the transition function $\mathcal{T}$ does not depend on the task variables. Moreover, assume the reward distribution is Gaussian with unit variance and Assumption 1 holds. Then, the log-pdf posterior function under the maximum entropy prior is given as:*

$$
\begin{aligned}
\forall \boldsymbol{\theta} \in \boldsymbol{\Theta}:\ \log \mu_{ME}\left(\boldsymbol{\theta} \mid \mathcal{H}_T\right) = &- \sum_{t=1}^{T} \sum_{h=1}^{H} \frac{1}{2}\left(r_h^t + \max_{a' \in \mathcal{A}} \mathbb{E}_{s'}\left[Q\left(s', a'\,;\, \boldsymbol{\theta}\right)\right] - Q\left(s_h^t, a_h^t\,;\, \boldsymbol{\theta}\right)\right)^2 \\
&+ \sum_{\tau \in \mathcal{D}_E} \alpha_\tau^\star \cdot \prod_{(s,a) \in \tau} \frac{\exp\left\{\beta \cdot Q\left(s, a\,;\, \boldsymbol{\theta}\right)\right\}}{\sum_{a' \in \mathcal{A}} \exp\left\{\beta \cdot Q\left(s, a'\,;\, \boldsymbol{\theta}\right)\right\}} + \textit{constant in } \boldsymbol{\theta},
\end{aligned}
\tag{22}
$$

where $\mathcal{H}_T = \left\{\left(\left(s_h^t, a_h^t, r_h^t, s_{h+1}^t\right)_{h=1}^{H}\right)_{t=1}^{T}\right\}$ is the history of online interactions, $\mathcal{D}_E$ is the expert demonstration data, β is the competence level of the expert in Assumption 1, and $\{\alpha_\tau^\star\}_{\tau \in \mathcal{D}_E}$ are derived from Proposition 1.

Remark. We note that, in principle, the ExPerior framework allows for task-dependent transition functions. In this case, the log-pdf in (22) provides an optimistic upper bound on the true posterior log-pdf function. See Hao et al. [23] for a similar analysis. We leave the general case for future work. Note that the second term of (22) is simply the log-pdf of the max-entropy prior.

Proof. Since the transition function is task-independent, the likelihood of an expert trajectory τ_{E} can be simplified as:

$$
\forall c \in \mathcal{C}: \quad m_{\tau_{\mathrm{E}}}(c) = \prod_{h=1}^{H} p_{\mathrm{E}}\left(a_h \mid s_h\,;\, c\right) \cdot \prod_{h=1}^{H} \mathcal{T}\left(s_{h+1} \mid s_h, a_h\right). \tag{23}
$$

The second term in (23) is constant in c. This implies that the likelihood function $m_{\tau_{\mathrm{E}}}(c)$ will depend on c only through the expert policy, which itself is a function of optimal Q-functions by Assumption 1. Note that the second term in the definition of $m_{\tau_{\mathrm{E}}}$ can be simply removed since we can re-weight the parameters $\boldsymbol{\alpha}$ in the optimization step (3) of Proposition 1. Hence, assuming the deep Q-network is expressive enough, without loss of generality, we can re-define the likelihood function of an expert trajectory $\tau_{\mathrm{E}} = (s_1, a_1, s_2, a_2, \ldots, s_H, a_H, s_{H+1})$ as

$$\forall \boldsymbol{\theta} \in \Theta: \quad m_{\tau_{\mathrm{E}}}(\boldsymbol{\theta}) = \prod_{h=1}^{H} \frac{\exp\left\{\beta \cdot Q\left(s_h, a_h\,;\,\boldsymbol{\theta}\right)\right\}}{\sum_{a' \in \mathcal{A}} \exp\left\{\beta \cdot Q\left(s_h, a'\,;\,\boldsymbol{\theta}\right)\right\}}.$$

We can now write the log-pdf of the posterior distribution of $\boldsymbol{\theta}$ given $\mathcal{H}_T$:

$$\begin{aligned}
\forall \boldsymbol{\theta} \in \Theta: \quad & \log \mu_{\mathrm{ME}}\left(\boldsymbol{\theta} \mid \mathcal{H}_T\right) \\
&= \log \mathrm{P}\left(\mathcal{H}_T \mid \boldsymbol{\theta}\right) + \log \mu_{\mathrm{ME}}(\boldsymbol{\theta}) + \text{constant in } \boldsymbol{\theta} \\
&= \sum_{t=1}^{L} \sum_{h=1}^{H} \log \rho\left(s_1^t\right) + \log R\left(r_h^t \,\middle|\, s_h^t, a_h^t\,;\,\boldsymbol{\theta}\right) + \log \mathcal{T}\left(s_{h+1}^t \,\middle|\, s_h^t, a_h^t\right) + \log \mu_{\mathrm{ME}}(\boldsymbol{\theta}) + \text{const.} \\
&= \sum_{t=1}^{L} \sum_{h=1}^{H} \log R\left(r_h^t \,\middle|\, s_h^t, a_h^t\,;\,\boldsymbol{\theta}\right) + \log \mu_{\mathrm{ME}}(\boldsymbol{\theta}) + \text{const.},
\end{aligned} \tag{24}$$

Now, given the Bellman equations, we can write the mean value of the reward function as

$$\forall s \in \mathcal{S}, a \in \mathcal{A}: \quad \mathbb{E}\left[R\left(s, a\,;\,\boldsymbol{\theta}\right)\right] = Q\left(s, a\,;\,\boldsymbol{\theta}\right) - \max_{a' \in \mathcal{A}} \mathbb{E}_{s'}\left[Q\left(s', a'\,;\,\boldsymbol{\theta}\right)\right]$$

The reward distribution is Gaussian with unit variance. Therefore,

$$\forall s \in \mathcal{S}, a \in \mathcal{A}, r \in \mathbb{R}: \quad R\left(r \mid s, a\,;\,\boldsymbol{\theta}\right) = \mathcal{N}\left(Q\left(s, a\,;\,\boldsymbol{\theta}\right) - \max_{a' \in \mathcal{A}} \mathbb{E}_{s'}\left[Q\left(s', a'\,;\,\boldsymbol{\theta}\right)\right], 1\right). \tag{25}$$

Moreover, by Proposition 1, the log-pdf of the maximum entropy expert prior is given as

$$\forall \boldsymbol{\theta} \in \Theta: \quad \log \mu_{\mathrm{ME}}(\boldsymbol{\theta}) = \sum_{\tau \in \mathcal{D}_{\mathrm{E}}} \alpha_\tau^\star \cdot m_\tau(\boldsymbol{\theta}) = \sum_{\tau \in \mathcal{D}_{\mathrm{E}}} \alpha_\tau^\star \cdot \prod_{(s,a) \in \tau} \frac{\exp\left\{\beta \cdot Q\left(s, a\,;\,\boldsymbol{\theta}\right)\right\}}{\sum_{a' \in \mathcal{A}} \exp\left\{\beta \cdot Q\left(s, a'\,;\,\boldsymbol{\theta}\right)\right\}}. \tag{26}$$

Combining (24) to (26), we conclude the proof. $\square$

A.6 Ensemble Marginal Likelihood

Proposition 3. *Consider a contextual MDP $\mathcal{M} = (\mathcal{S}, \mathcal{A}, \mathcal{T}, R, H, \rho, \mu^\star)$. Assume the transition function $\mathcal{T}$ does not depend on the task variables and Assumption 1 holds. Then, the negative marginal log-likelihood of expert data $\mathcal{D}_E$ under the ensemble prior $\mu_{\boldsymbol{\theta}_{\mathrm{ens}}}$ is upper bounded by*

$$-\log \mathrm{P}_E\left(\mathcal{D}_E\,;\,\mu_{\boldsymbol{\theta}_{\mathrm{ens}}}\right) \leq \frac{1}{L} \sum_{i=1}^{L} \sum_{\tau \in \mathcal{D}_E} \sum_{(s,a) \in \tau} \log\left(\sum_{a' \in \mathcal{A}} \exp\left\{\beta \cdot Q\left(s, a'\,;\,\boldsymbol{\theta}_{\mathrm{ens}}^i\right)\right\}\right) - \beta \cdot Q\left(s, a\,;\,\boldsymbol{\theta}_{\mathrm{ens}}^i\right),$$

where β is the competence level of the expert in Assumption 1.

Proof. Recalling (1), the log-likelihood of the expert trajectories $\mathcal{D}_{\mathrm{E}}$ under $\mu_{\boldsymbol{\theta}_{\mathrm{ens}}}$ is given by

$$
\begin{aligned}
-\log \mathrm{P}_{\mathrm{E}}\left(\mathcal{D}_{\mathrm{E}}\,;\,\mu_{\boldsymbol{\theta}_{\mathrm{ens}}}\right) &= \sum_{\tau^{(i)} \in \mathcal{D}_{\mathrm{E}}} -\log \mathbb{E}_{\boldsymbol{\theta} \sim \mu_{\boldsymbol{\theta}_{\mathrm{ens}}}}\left[\rho(s_1^{(i)}) \prod_{h=1}^{H} p_{\mathrm{E}}\left(a_h^{(i)} \,\middle|\, s_h^{(i)}\,;\,\boldsymbol{\theta}\right) \mathcal{T}\left(s_{h+1}^{(i)} \,\middle|\, s_h^{(i)}, a_h^{(i)}\right)\right] \\
&= \sum_{\tau^{(i)} \in \mathcal{D}_{\mathrm{E}}} -\log \mathbb{E}_{\boldsymbol{\theta} \sim \mu_{\boldsymbol{\theta}_{\mathrm{ens}}}}\left[\prod_{h=1}^{H} p_{\mathrm{E}}\left(a_h^{(i)} \,\middle|\, s_h^{(i)}\,;\,\boldsymbol{\theta}\right)\right] + \text{constant in } \boldsymbol{\theta}_{\mathrm{ens}} \\
&\hspace{20em} (\rho, \mathcal{T} \text{ do not depend on } \boldsymbol{\theta}) \\
&= \sum_{\tau^{(i)} \in \mathcal{D}_{\mathrm{E}}} -\log\left(\frac{1}{L}\sum_{j=1}^{L}\prod_{h=1}^{H} p_{\mathrm{E}}\left(a_h^{(i)} \,\middle|\, s_h^{(i)}\,;\,\boldsymbol{\theta}_{\mathrm{ens}}^{j}\right)\right) \\
&\hspace{20em} (\text{By Definition of } \mu_{\boldsymbol{\theta}_{\mathrm{ens}}}) \\
&\leq \sum_{\tau^{(i)} \in \mathcal{D}_{\mathrm{E}}} \frac{1}{L}\sum_{j=1}^{L}\sum_{h=1}^{H} -\log p_{\mathrm{E}}\left(a_h^{(i)} \,\middle|\, s_h^{(i)}\,;\,\boldsymbol{\theta}_{\mathrm{ens}}^{j}\right) \qquad \text{By Jensen's inequality} \\
&= \frac{1}{L}\sum_{i=1}^{L}\sum_{\tau \in \mathcal{D}_{\mathrm{E}}}\sum_{(s,a)\in\tau}\left[\log\left(\sum_{a' \in \mathcal{A}} \exp\left\{\beta \cdot Q\left(s, a'\,;\,\boldsymbol{\theta}_{\mathrm{ens}}^{i}\right)\right\}\right) - \beta \cdot Q\left(s, a\,;\,\boldsymbol{\theta}_{\mathrm{ens}}^{i}\right)\right] \\
&\hspace{20em} \text{By Assumption 1}
\end{aligned}
$$

□

B High-Level Implementation of ExPerior

Algorithm 2 Max-Entropy Posterior Sampling (ExPerior)

Input: Expert demonstrations $\mathcal{D}_{\mathrm{E}}$, Reference distribution μ_0, $\lambda^\star \geq 0$, and unknown task $c \sim \mu^\star$.
$\mu_{\mathrm{ME}} \leftarrow \textsc{MaxEntropyExpertPrior}(\mu_0, \mathcal{D}_{\mathrm{E}}, \lambda^\star)$
history $\leftarrow \{\}$
for episode $t \leftarrow 1, 2, \ldots$ **do**
  sample $c_t \sim \mu_{\mathrm{ME}}\left(\cdot \mid \textit{history}\right)$ // posterior sampling
  for timestep $h \leftarrow 1, 2, \ldots, H$ **do**
    take action $a_h^t \sim \pi_{c_t}\left(\cdot \mid s_h\right)$
    observe $r_h^t \sim R(s_h^t, a_h^t, c)$, $s_{h+1}^t \sim \mathcal{T}\left(s_h^t, a_h^t, c\right)$ and append $(a_h^t, r_h^t, s_{h+1}^t)$ to *history*
  end for
end for

C Additional Experiments

C.1 Ablation Studies for Bernoulli Mult-Armed Bandits

Table 2: Ablation experiments to assess the robustness of ExPerior to misspecified expert models. Random-optimal experts choose the optimal action with probability γ and choose random actions with probability $1 - \gamma$. `ExPerior-MaxEnt` achieves consistent out-performance by setting the hyperparameter $\beta = 10$. while `ExPerior-Param` get almost similar results for $\beta = 1$ and $\beta = 2.5$.

	Optimal	Noisily-Rational				Random-Optimal			
		$\beta = 0.1$	$\beta = 1$	$\beta = 2.5$	$\beta = 10$	$\gamma = 0.0$	$\gamma = 0.25$	$\gamma = 0.5$	$\gamma = 0.75$
$\beta = 0.1$									
ExPerior-MaxEnt	51.7 ± 5.1	52.3 ± 5.3	52.3 ± 5.3	52.0 ± 5.1	51.7 ± 5.0	52.3 ± 5.3	52.1 ± 5.1	52.0 ± 5.1	51.8 ± 5.0
ExPerior-Param	11.1 ± 4.3	33.1 ± 7.3	12.6 ± 3.5	11.7 ± 3.8	10.9 ± 4.2	40.1 ± 9.6	12.3 ± 4.7	11.4 ± 4.0	10.7 ± 4.2
$\beta = 1$									
ExPerior-MaxEnt	45.7 ± 3.4	52.2 ± 5.3	51.6 ± 5.1	50.0 ± 4.8	47.3 ± 3.8	52.5 ± 5.3	51.0 ± 4.8	49.1 ± 4.2	48.0 ± 3.6
ExPerior-Param	9.1 ± 3.0	21.3 ± 1.3	13.4 ± 2.9	10.1 ± 3.0	9.4 ± 3.1	22.8 ± 1.3	9.8 ± 3.0	8.6 ± 2.7	8.8 ± 2.9
$\beta = 2.5$									
ExPerior-MaxEnt	37.0 ± 1.9	52.1 ± 5.3	51.0 ± 4.9	47.1 ± 4.5	38.3 ± 2.0	52.1 ± 5.1	48.9 ± 4.1	44.8 ± 3.2	40.5 ± 2.1
ExPerior-Param	8.5 ± 2.8	24.3 ± 1.2	19.0 ± 2.1	12.8 ± 2.9	9.2 ± 3.1	24.6 ± 1.2	15.9 ± 3.0	10.9 ± 3.2	8.8 ± 2.9
$\beta = 10$									
ExPerior-MaxEnt	38.5 ± 9.4	52.0 ± 5.2	47.6 ± 4.4	39.7 ± 2.9	29.7 ± 3.6	52.5 ± 5.3	41.9 ± 2.6	37.7 ± 2.8	31.9 ± 3.0
ExPerior-Param	11.2 ± 4.8	26.9 ± 1.2	25.0 ± 1.5	21.0 ± 2.1	11.8 ± 3.3	26.8 ± 1.1	23.2 ± 1.8	20.1 ± 2.5	16.1 ± 3.0
Oracle-TS	8.5 ± 2.7	8.5 ± 2.7	8.5 ± 2.7	8.5 ± 2.7	8.5 ± 2.7	8.5 ± 2.7	8.5 ± 2.7	8.5 ± 2.7	8.5 ± 2.7
Oracle-TS (SGLD)	24.2 ± 3.9	24.2 ± 3.9	24.2 ± 3.9	24.2 ± 3.9	24.2 ± 3.9	24.2 ± 3.9	24.2 ± 3.9	24.2 ± 3.9	24.2 ± 3.9

Table 3: Superiority of `ExPerior-MaxEnt` compared to `ExPerior-Param` with misspecified parametric prior.

	Low Entropy	Mid-Entropy	High-Entropy
ExPerior-Param	0.7 ± 0.3	6.8 ± 0.8	24.5 ± 2.8
ExPerior-MaxEnt	11.6 ± 1.3	25.7 ± 1.2	41.3 ± 2.2
ExPerior-Param (Gamma)	39.3 ± 2.2	36.8 ± 0.9	51.8 ± 3.6
ExPerior-Param (Beta-SGLD)	60.2 ± 6.3	40.4 ± 2.0	45.6 ± 2.0
ExPerior-Param (Normal)	546.5 ± 153.4	492.5 ± 185.6	461.8 ± 104.8
Oracle-TS	0.9 ± 0.4	7.3 ± 0.8	21.5 ± 2.2
Oracle-TS (SGLD)	11.0 ± 1.6	21.2 ± 1.0	39.9 ± 3.2

C.2 Frozen Lake

Table 4: The average reward per episode in Frozen Lake (MDP) after 90,000 training steps.

	Fixed # Hazard = 9				Fixed $\beta = 1$			
	$\beta = 0.1$	$\beta = 1$	$\beta = 2.5$	$\beta = 10$	# Hazard = 2	# Hazard = 5	# Hazard = 7	# Hazard = 9
(MDP)								
ExPerior-MaxEnt	-23.36 $\pm$ 1.26	12.26 $\pm$ 0.29	12.68 $\pm$ 0.03	**12.71 $\pm$ 0.03**	**13.02 $\pm$ 0.18**	**12.78 $\pm$ 0.11**	**12.78 $\pm$ 0.06**	12.26 $\pm$ 0.29
ExPerior-Param	-25.53 $\pm$ 2.35	**12.64 $\pm$ 0.08**	**12.70 $\pm$ 0.03**	12.68 $\pm$ 0.03	13.00 $\pm$ 0.18	**12.78 $\pm$ 0.12**	12.73 $\pm$ 0.07	**12.64 $\pm$ 0.08**
Naïve Boot-DQN	-23.32 $\pm$ 0.69	-23.32 $\pm$ 0.69	-23.32 $\pm$ 0.69	-23.32 $\pm$ 0.69	-14.39 $\pm$ 5.22	-20.99 $\pm$ 2.86	-20.39 $\pm$ 1.75	-23.32 $\pm$ 0.69
ExPLORe	**11.74 $\pm$ 0.41**	11.75 $\pm$ 0.63	11.96 $\pm$ 0.28	12.3 $\pm$ 0.22	-113.84 $\pm$ 17.50	-54.89 $\pm$ 13.75	-10.00 $\pm$ 7.60	11.75 $\pm$ 0.63
Optimal	12.71 $\pm$ 0.03	12.71 $\pm$ 0.03	12.71 $\pm$ 0.03	12.71 $\pm$ 0.03	13.02 $\pm$ 0.18	12.78 $\pm$ 0.11	12.76 $\pm$ 0.06	12.64 $\pm$ 0.03