

---

# Offline Imitation Learning in $Q^\pi$ -Realizable MDPs without Expert Realizability

---

Anonymous Author(s)

Affiliation

Address

email

## Abstract

1 We study the problem of offline imitation learning in Markov decision processes  
2 (MDPs), where the goal is to learn a well-performing policy given a dataset of  
3 state-action pairs generated by an expert policy. Complementing a recent line of  
4 work on this topic that assumes that the expert policy belongs to a tractable class of  
5 known policies, we approach this problem from a new angle and leverage another  
6 type of structural assumption about the environment. Specifically, for the class  
7 of linear  $Q^\pi$ -realizable MDPs, we introduce a new algorithm called saddle-point  
8 offline imitation learning (SPOIL), which is guaranteed to match the performance of  
9 any expert up to an additive error  $\varepsilon$  with access to  $\mathcal{O}(\varepsilon^{-2})$  samples. Moreover, we  
10 extend this result to possibly non-linear  $Q^\pi$ -realizable MDPs at the cost of a worse  
11 sample complexity of order  $\mathcal{O}(\varepsilon^{-4})$ . Finally, our analysis suggests a new loss  
12 function for training critic networks from expert data in deep imitation learning.  
13 Empirical evaluations on standard benchmarks demonstrate that the neural net  
14 implementation of SPOIL is superior to behavior cloning and competitive with  
15 state-of-the-art algorithms.

## 16 1 Introduction

17 In imitation learning (IL), a learner observes a finite dataset of state-action pairs generated by  
18 an expert policy interacting with an environment modeled as a Markov Decision Process (MDP;  
19 [Puterman \[1994\]](#)). The learner’s objective is to find a policy that performs nearly as well as the  
20 expert policy with respect to an unknown ground-truth reward function. This work focuses on *offline*  
21 *imitation learning*, where the learner cannot collect new state-action sequences from the MDP used  
22 for generating the expert’s data and proposes new algorithms for this setting under a previously  
23 under-explored set of structural assumptions on the learning environment.

24 Recent years saw a quite significant surge of interest in the problem of imitation learning, not unlikely  
25 due to its relevance to next-token prediction in generative language models [[Rajaraman et al., 2020](#),  
26 [Foster et al., 2024](#), [Rohatgi et al., 2025](#)]. A common feature of these recent works is that they all  
27 make the assumption that the expert data has been generated by a fixed policy that belongs to a  
28 known, finite class of policies and they return policies within the same class. Several clean and  
29 elegant results were proved under this assumption, in particular showing the existence of conceptually  
30 simple algorithms achieving tight upper bounds on the sample complexity of finding good solutions,  
31 and lower bounds demonstrating the near-optimality of these algorithms under said assumptions.  
32 These bounds typically depend on a measure of complexity of the policy class (as measured by,  
33 say, its covering number). However, further scrutiny reveals that these assumptions may not always  
34 be verified or even reasonable: in many cases of significant practical interest, there is no reason to  
35 believe that the expert policy may be easily modeled within a simple and tractable policy class. For  
36 instance, in the popular use case of learning from human feedback, it is arguably quite unlikely that

data would be generated in a consistent, systematically predictable way that can be modeled as a simple policy mapping states to actions. Indeed, human behavior can be nonstationary, irrational, or even be influenced by unobserved confounders not captured by the state representation. We address these limitations by exploring an alternative framework for imitation learning, which reasons about the structure of the *value functions* of the policies used by the *learning algorithm* itself, as opposed to making assumptions about the structure of the policy followed by the expert.

A prevalent assumption in existing analyses of offline imitation learning algorithms [Rajaraman et al., 2020, Foster et al., 2024, Rohatgi et al., 2025] is expert realizability.

**Assumption** (Expert realizability). *The learner has access to a function class  $\Pi^E$  that contains the unknown expert policy  $\pi_E$ , that is, such that  $\pi_E \in \Pi^E$ .*

This assumption can be unreasonable for complex expert policies. Furthermore, the sample complexity guarantess in Rajaraman et al. [2020], Foster et al. [2024], Rohatgi et al. [2025] scale with  $\log |\Pi^E|$  (assuming  $\Pi^E$  is finite), meaning large policy classes, potentially necessary to realize the expert, lead to deteriorated guarantees. Additionally, the consequences of misspecification, *i.e.*  $\pi_E \notin \Pi^E$ , are often severe. For instance, Rohatgi et al. [2025] demonstrated that if the policy class  $\Pi^E$  is misspecified, then it is computationally intractable to learn  $\arg \min_{\pi \in \Pi^E} \mathcal{D}_H^2(\mathbb{P}^\pi, \mathbb{P}^{\pi_E})$ , the best in-class policy under the Hellinger distance, in an offline manner. However, this theoretical intractability under misspecification seems at odds with practical scenarios, such as training large language models via next-token prediction (a form of offline IL), which perform well despite the expert policy (derived from human-written text) likely not belonging to any reasonable policy class  $\Pi^E$ .

To address this apparent discrepancy, we initiate the study of offline imitation learning by leveraging structural assumptions about the MDP rather than relying on expert realizability. For example, in language tasks, structural assumptions might involve deterministic, tree-shaped MDPs. In robotics, one might assume that next states are determined by compact feature representations of current state-action pairs. More generally, we consider MDPs where the action-value function of any policy can be written as a linear combination of features known to the learner. Such MDPs are referred to as linear  $Q^\pi$ -realizable MDPs, a class that has been central to recent works in reinforcement learning theory [Weisz et al., 2023, Mhammedi, 2024, Tkachuk et al., 2024]. Our primary contribution is to show that, for this class of MDPs, it is possible to develop algorithms that guarantee to output a policy performing arbitrarily close to the expert policy *without imposing expert realizability*.

The algorithm is based on a simple primal-dual formulation of the problem of imitation learning, which characterizes the solution as the saddle-point of a convex-concave objective function. The primal variables correspond to policies in the MDP and the dual variables to  $Q$ -functions, which motivates a very simple saddle-point optimization algorithm for imitation learning: in a sequence of rounds, the primal player (the *actor*) picks a policy and the dual player (the *critic*) picks a  $Q$ -function, respectively trying to minimize and maximize the objective. We accordingly call the method SP0IL, standing for Saddle-Point Offline Imitation Learning. In the case of linear function approximation, both update steps of SP0IL can be performed very efficiently (in time linear in the feature dimension). For general function approximation, the  $Q$ -function updates can be performed by solving a simple linear optimization problem, which is straightforward to solve in practical scenarios. When instantiated with neural networks, empirical experiments show its performance is competitive with (and in some cases superior to, *e.g.*, behavior cloning) state-of-the-art offline imitation learning algorithms. Interestingly, our algorithm shares a good degree of similarity with the state-of-the-art method of Garg et al. [2021] called IQ-Learn, which is also derived from a primal-dual perspective. We discuss these similarities in depth and argue that SP0IL provides a superior solution to the IQ-Learn objective (at least inasmuch as it is more amenable to theoretical analysis).

To the best of our knowledge, this is the first result showing that leveraging structural assumptions of the underlying MDP can guarantee matching the expert performance as the number of expert transitions goes to infinity without imposing any form of expert realizability assumption. For clarity, we compare our contribution with existing results in Table 1.

**Notation.** We use  $\Delta(\mathcal{Z})$  to denote the simplex over the countable set  $\mathcal{Z}$ . Given two probability distributions  $p, q \in \Delta(\mathcal{Z})$ , we denote the Kullback-Leibler divergence as  $\mathcal{D}_{\text{KL}}(p, q) = \sum_{z \in \mathcal{Z}} p(z) \log \frac{p(z)}{q(z)}$ . We denote  $\langle \cdot, \cdot \rangle$  the inner product between two finite-dimensional vectors, and  $\|\cdot\|$  the Euclidean norm. We denote  $\mathcal{U}([K])$  the uniform distribution over the set  $[K] = \{1, \dots, K\}$ . The Euclidean ball of radius  $R > 0$  centered at the origin is denoted as  $\mathfrak{B}(R)$ .

Table 1: Comparison with related algorithms. We denoted the class of deterministic linear experts as  $\Pi_{\text{det,lin}}^E = \{\pi : \exists \theta \in \mathfrak{B}(B_\theta), \pi(\cdot) = \arg \max_{a \in \mathcal{A}} \langle \theta, \varphi(\cdot, a) \rangle\}$ , and an arbitrary policy class as  $\Pi^E$ . We also define  $W = \max_{\pi \in \Pi^E, (x,a) \in \mathcal{X} \times \mathcal{A}, h \in [H]} \frac{\pi_{E,h}(a|x)}{\pi_h(a|x)}$  and  $\epsilon_{\text{miss}} = \min_{\pi \in \Pi^E} \mathcal{D}_H^2(\mathbb{P}^\pi, \mathbb{P}^{\pi_E})$ .

Algorithm	Structural assumptions	Avoids expert realizability	Infinite horizon	Expert class	Expert Traj. ( $\tau_E$ )
BC with log loss [Foster et al., 2024]	None	$\times$	$\times$	$\Pi^E$	$\mathcal{O}\left(\frac{H^2 \log  \Pi^E }{\epsilon^2}\right)$
BC with 0-1 loss [Rajaraman et al., 2021]	None	$\times$	$\times$	$\Pi_{\text{det,lin}}^E$	$\tilde{\mathcal{O}}\left(\frac{H^2 d}{\epsilon}\right)$
BoostedLogLossBC [Rohatgi et al., 2025]	None	$\checkmark$ with a misspecification error of $\tilde{\mathcal{O}}(H \log(W) \epsilon_{\text{miss}})$	$\times$	$\Pi^E$	$\mathcal{O}\left(\frac{H^2 \log  \Pi^E }{\epsilon^2}\right)$
Projection [Abbeel and Ng, 2004]	Linear reward Known transitions	$\checkmark$	$\checkmark$	—	$\tilde{\mathcal{O}}\left(\frac{d}{(1-\gamma)^2 \epsilon^2}\right)$
MWAL [Syed and Schapire, 2007]	Linear reward Known transitions	$\checkmark$	$\checkmark$	—	$\tilde{\mathcal{O}}\left(\frac{\log(d)}{(1-\gamma)^2 \epsilon^2}\right)$
SPOIL (Ours)	Linear $Q^\pi$ -realizability	$\checkmark$	$\checkmark$	—	$\tilde{\mathcal{O}}\left(\frac{d}{(1-\gamma)^2 \epsilon^2}\right)$
SPOIL (Ours)	$Q^\pi$ -realizability	$\checkmark$	$\checkmark$	—	$\tilde{\mathcal{O}}\left(\frac{\log C_\pi(Q)}{(1-\gamma)^6 \epsilon^4}\right)$

## 2 Preliminaries

We begin by introducing the problem of offline imitation learning in discounted MDPs together with the assumptions we will consider throughout the paper.

**Markov decision processes.** We formalize the learning problem in a discounted MDP  $\mathcal{M} = (\mathcal{X}, \mathcal{A}, r, P, \gamma, \nu_0)$ , where  $\mathcal{X}$  is the state space which we assume finite but too large to be enumerated,  $\mathcal{A}$  is a finite action space with  $A$  actions,  $r : \mathcal{X} \times \mathcal{A} \rightarrow [0, 1]$  is the unknown reward function,  $P : \mathcal{X} \times \mathcal{A} \rightarrow \Delta(\mathcal{X})$  is the unknown transition kernel,  $\gamma \in [0, 1)$  is the discount factor, and  $\nu_0 \in \Delta(\mathcal{X})$  is the initial state distribution. For any state-action-state triplet  $(x, a, x')$ ,  $P(x' | x, a)$  denotes the probability of landing in state  $x'$  after taking action  $a$  in state  $x$ . A *stationary policy* (or simply *policy*)  $\pi : \mathcal{X} \rightarrow \Delta(\mathcal{A})$  is a mapping from states to distributions over actions. The interaction of a policy  $\pi$  with the environment  $\mathcal{M}$  unfolds as follows: an initial state  $X_0 \sim \nu_0$  is drawn, and for each subsequent time step  $h \geq 0$ , an action  $A_h \sim \pi(\cdot | X_h)$  is taken, a reward  $r(X_h, A_h)$  is received, and the agent transitions to a new state  $X_{h+1} \sim P(\cdot | X_h, A_h)$ . We denote  $\mathbb{P}^\pi$  the resulting probability distribution over trajectories, and  $\mathbb{E}^\pi$  the corresponding expectation operator. For any state  $x \in \mathcal{X}$ , we define the state value function of the policy  $\pi$  as  $V^\pi(x) = \mathbb{E}^\pi[\sum_{h=0}^{\infty} \gamma^h r(X_h, A_h) | X_0 = x]$ . Analogously, we define the state-action value function as  $Q^\pi(x, a) = \mathbb{E}^\pi[\sum_{h=0}^{\infty} \gamma^h r(X_h, A_h) | X_0 = x, A_0 = a]$ . The value functions are tied together via the *Bellman equations*

$$V^\pi(x) = \sum_{a \in \mathcal{A}} \pi(a|x) Q^\pi(x, a), \quad \text{and} \quad Q^\pi(x, a) = r(x, a) + \gamma \sum_{x' \in \mathcal{X}} P(x' | x, a) V^\pi(x').$$

Additionally, we will sometimes use the notation  $Q(x, \pi)$  to denote  $\sum_a \pi(a|x) Q(x, a)$  for any policy  $\pi$  and any function  $Q : \mathcal{X} \times \mathcal{A} \rightarrow \mathbb{R}$ . Note that this notation allows us to write  $V^\pi(x) = Q^\pi(x, \pi)$ . Any policy  $\pi$  induces an *occupancy measure*  $\mu^\pi \in \Delta(\mathcal{X} \times \mathcal{A})$  over state-action pairs, defined as the discounted total expected times that each state-action pair is visited by policy  $\pi$ . The same quantity defined for states is called the state-occupancy measure and is denoted as  $\nu^\pi \in \Delta(\mathcal{X})$ . For any state-action pair  $(x, a) \in \mathcal{X} \times \mathcal{A}$ , they are respectively defined as

$$\nu^\pi(x) = (1 - \gamma) \sum_{h=0}^{\infty} \gamma^h \mathbb{P}^\pi[X_h = x], \quad \text{and} \quad \mu^\pi(x, a) = (1 - \gamma) \sum_{h=0}^{\infty} \gamma^h \mathbb{P}^\pi[X_h = x, A_h = a],$$

and they are related to each other by the *flow conditions* (sometimes called “Bellman flow conditions”)

$$\nu^\pi(x) = \gamma \sum_{x', a'} P(x|x', a') \mu^\pi(x', a') + (1 - \gamma) \nu_0(x). \quad (1)$$

Notably, these definitions and the flow conditions remain valid for general history-dependent policies  $\pi$  that may take the entire history of state-action pairs  $(X_1, A_1, \dots, X_h)$  into account when selecting each action  $A_h$ . Finally, we let  $\rho^\pi = \mathbb{E}^\pi [\sum_{h=0}^{\infty} \gamma^h r(X_h, A_h)]$  stand for the total expected return of a (potentially nonstationary) policy  $\pi$ . The following useful result, commonly called the *performance-difference lemma* (Kakade and Langford 2002, see also Eq. 7.14 in Howard 1960), gives a useful expression for the performance gap between two policies.

**Lemma 1.** *Let  $\pi$  be a stationary policy and  $\pi'$  be any policy. Then,*

$$\rho^{\pi'} - \rho^\pi = \mathbb{E}_{(X,A) \sim \mu^{\pi'}} [Q^\pi(X, A) - V^\pi(X)].$$

Note that this lemma is generally stated for stationary policies, but we will find it useful later to use it with general history-dependent policies. We provide the straightforward proof in Appendix B.

**Imitation Learning.** We consider the problem of offline imitation learning. Given a dataset  $\mathcal{D}^{\pi_E} = \{X_E^i, A_E^i\}_{i=1}^{\tau_E}$  of state-action pairs sampled from an expert policy's occupancy measure  $\mu^{\pi_E}$ , our objective is to design an algorithm, Alg, that produces a policy  $\pi^{\text{out}}$  satisfying

$$\mathbb{E} [\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] \leq \varepsilon. \quad (2)$$

The algorithm is not allowed any further interaction with the expert policy or the MDP  $\mathcal{M}$  and only has to work with the record of state-action pairs contained in the data set. As stated in the introduction, we aim to achieve this *without imposing expert realizability*. Instead, we consider the following structural assumption on the environment.

**Assumption 1** (Linear  $Q^\pi$ -realizability). *The action value function of any policy  $\pi$  can be written as a linear combination of known features. That is, there exists a known mapping  $\varphi : \mathcal{X} \times \mathcal{A} \rightarrow \mathbb{R}^d$  such that for any policy  $\pi$ , there exists a vector  $\theta^\pi \in \mathbb{R}^d$  such that for any state-action pair  $(x, a)$ ,  $Q^\pi(x, a) = \langle \varphi(x, a), \theta^\pi \rangle$ . Moreover, we assume  $\|\theta^\pi\| \leq B_\theta$  for all  $\pi$ , and  $\sup_{x,a} \|\varphi(x, a)\| \leq B_\varphi$ .*

We will also consider the general function approximation setting, where the action value function of any policy  $\pi$  can be represented by some function class  $\mathcal{Q} \subset \mathbb{R}^{\mathcal{X} \times \mathcal{A}}$ .

**Assumption 2** ( $Q^\pi$ -realizability). *An MDP is said  $Q^\pi$ -realizable if there exists a function class  $\mathcal{Q} \subset \mathbb{R}^{\mathcal{X} \times \mathcal{A}}$  such that for any policy  $\pi$ , it holds that  $Q^\pi \in \mathcal{Q}$ , and for any  $Q \in \mathcal{Q}$ ,  $\|Q\|_\infty \leq \frac{1}{1-\gamma}$ .*

For this assumption to make sense, we typically require the function class  $\mathcal{Q}$  to have bounded capacity. We formalize this via covering numbers, defined as follows.

**Definition 1** (Covering number). *Let  $(M, d)$  be a metric space,  $K$  be a subset of  $M$ , and  $\epsilon > 0$ . A set  $\mathcal{C}_\epsilon(K, d)$  is an  $\epsilon$ -covering of  $K$  if for any  $x \in K$ , there exists  $y \in \mathcal{C}_\epsilon(K, d)$  such that  $d(x, y) \leq \epsilon$ . The covering number of  $K$ ,  $\mathcal{N}_\epsilon(K, d)$ , is the minimum cardinality of any such covering of  $K$ .*

### 3 Primal-dual offline imitation learning

In order to introduce our main algorithmic idea, we define the following objective function:

$$\mathcal{L}(\pi; Q) = \mathbb{E}_{(X,A) \sim \mu^{\pi_E}} [Q(X, A) - Q(X, \pi)],$$

where we denoted  $Q(X, \pi) = \mathbb{E}_{A' \sim \pi(\cdot|X)} [Q(X, A')]$ . Our main observation is that the main objective function we consider can be rewritten in terms of this function as follows:

$$\rho^{\pi_E} - \rho^\pi = \mathcal{L}(\pi; Q^\pi) \leq \sup_{Q \in \mathcal{Q}} \mathcal{L}(\pi; Q).$$

This suggests that a good policy  $\pi^{\text{out}}$  may be found by solving the *saddle-point* problem  $\min_\pi \sup_{Q \in \mathcal{Q}} \mathcal{L}(\pi; Q)$ . Indeed, if one is able to produce a policy  $\pi^{\text{out}}$  that satisfies  $\sup_{Q \in \mathcal{Q}} \mathcal{L}(\pi^{\text{out}}; Q) \leq \varepsilon$ , then the above inequality implies that the suboptimality of  $\pi^{\text{out}}$  as compared to  $\pi_E$  will also be at most  $\varepsilon$ .

Inspired by this observation, we set out to design an incremental *primal-dual* optimization algorithm to approximate the saddle point of the function  $\mathcal{L}$ . In each iteration  $k = 1, 2, \dots, K$ , the algorithm performs two updates: a primal update that corresponds to policy updates aiming to minimize  $\mathcal{L}$ , and a dual update that computes action-value function estimates and aims to maximize  $\mathcal{L}$ . Following a

157 common terminology in reinforcement learning, we will sometimes refer to the primal updates as  
 158 *actor* updates and the dual updates as *critic* updates.

159 In order to turn these insights into a practical algorithm, we define the following empirical estimate  
 160 of the objective function  $\mathcal{L}$ :

$$\widehat{\mathcal{L}}(\pi; Q) = \frac{1}{\tau_E} \sum_{i=1}^{\tau_E} (Q(X_E^i, A_E^i) - Q(X_E^i, \pi)).$$

161 For a fixed  $Q$  and  $\pi$ , this is clearly an unbiased estimator of  $\mathcal{L}$ . In line with the derivations above, we  
 162 choose our critic and actor updates respectively as

$$Q_k \in \arg \max_{Q \in \mathcal{Q}} \widehat{\mathcal{L}}(\pi_k; Q), \quad \text{and} \quad \pi_{k+1}(a|x) = \frac{\pi_k(a|x) e^{\eta Q_k(x,a)}}{\sum_{a'} \pi_k(a'|x) e^{\eta Q_k(x,a')}},$$

163 where  $\eta > 0$  is a *learning-rate* (or *stepsize*) parameter that modulates the strength of the policy  
 164 updates. After performing  $K$  updates, the algorithm chooses a random index  $I$  uniformly on the  
 165 integers in  $\llbracket 1, K \rrbracket$ , and returns  $\pi^{\text{out}} = \pi_I$ . We refer to this algorithm as Saddle-Point Offline Imitation  
 166 Learning (SP0IL). This algorithm design is justified by the following simple error decomposition  
 167 that lies at the heart of our main results.

168 **Proposition 1.** *Let  $\Delta(\pi) = \mathbb{E} \left[ \sup_{Q \in \mathcal{Q}} |\mathcal{L}(\pi; Q) - \widehat{\mathcal{L}}(\pi; Q)| \right]$ . The output of SP0IL satisfies*

$$\mathbb{E} [\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] \leq \frac{1}{K} \sum_{k=1}^K \mathbb{E} [\mathcal{L}(\pi_k; Q_k)] + \frac{2}{K} \sum_{k=1}^K \mathbb{E} [\Delta(\pi_k)].$$

169 *Proof.* The proof simply follows by noticing

$$\begin{aligned} \mathbb{E} [\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] &= \frac{1}{K} \sum_{k=1}^K \mathbb{E} [\mathcal{L}(\pi_k; Q^{\pi_k})] \leq \frac{1}{K} \sum_{k=1}^K \mathbb{E} [\widehat{\mathcal{L}}(\pi_k; Q^{\pi_k})] + \frac{1}{K} \sum_{k=1}^K \mathbb{E} [\Delta(\pi_k)] \\ &\leq \frac{1}{K} \sum_{k=1}^K \mathbb{E} [\widehat{\mathcal{L}}(\pi_k; Q_k)] + \frac{1}{K} \sum_{k=1}^K \mathbb{E} [\Delta(\pi_k)] \leq \frac{1}{K} \sum_{k=1}^K \mathbb{E} [\mathcal{L}(\pi_k; Q_k)] + \frac{2}{K} \sum_{k=1}^K \mathbb{E} [\Delta(\pi_k)], \end{aligned}$$

170 where we have used the definitions of  $\Delta$  and  $Q_k$  in the first and second lines, respectively.  $\square$

171 The first term in this decomposition corresponds to the *regret* of the policy player  $\pi$  against the  
 172 comparator strategy  $\pi_E$  and can be controlled with probability 1 via standard tools of online learning  
 173 (as found in the excellent books of [Cesa-Bianchi and Lugosi 2006](#) and [Orabona 2023](#)). The second  
 174 term measures the estimation error of the objective function  $\mathcal{L}$  uniformly over the space of action-  
 175 value functions  $\mathcal{Q}$  and along the policies played by the algorithm, and can be controlled via standard  
 176 concentration arguments. Altogether, the proposition suggests that SP0IL will return a good policy if  
 177 these estimation errors can be bounded reasonably—a fact we will formally show in the next section.

178 Before stating our performance guarantees for the concrete settings we consider in this paper, we  
 179 pause to point out a peculiar connection between the algorithm described above and the inverse  
 180 Q-learning (IQ-Learn) algorithm of [Garg et al. \[2021\]](#). While motivated using completely different  
 181 arguments, the saddle-point objective function optimized by IQ-Learn is nearly identical to our  
 182 function  $\mathcal{L}$ : after removing entropy-regularization and setting their reward regularizer  $\psi$  to zero, one  
 183 can verify using the flow constraint (Eq. 1) that their function  $\mathcal{J}$  is *identical* to our  $\mathcal{L}$ . Ultimately,  
 184 [Garg et al. \[2021\]](#) draw different conclusions from this saddle-point formulation, and propose to solve  
 185 it by computing  $\pi_Q = \arg \min_{\pi} \mathcal{J}(Q)$  and optimize the *dual function*  $g(Q) = \min_{\pi} \mathcal{L}(\pi; Q)$ . This  
 186 function, however, can be highly nonsmooth and difficult to optimize, which is why IQ-Learn needs  
 187 to heavily rely on regularization both in  $\pi$  and  $Q$ . In contrast, our algorithm can be seen as trying to  
 188 optimize the *primal function*  $f(\pi) = \max_Q \mathcal{L}(\pi; Q)$  in terms of the policy  $\pi$ , which can be done in a  
 189 stable way by incremental policy updates. Additionally, as Proposition 1 clearly reveals, optimizing  
 190 the primal objective allows us to directly reason about the performance of the output policy. In  
 191 contrast, we do not see a clear way to do this for the dual objective optimized by IQ-Learn.

192 In what follows, we instantiate SP0IL in two settings of particular interest, depending on the Q-  
 193 function class being used. We first provide a set of results for linear function approximation (where  
 194 the algorithm is very easy to implement and analyze) and for general function classes (where  
 195 implementation and analysis are both less straightforward).



**Algorithm 1** SPOIL with linear FA

**Input:** Number of expert trajectories  $\tau_E$ , learning rate  $\eta$ , number of iterations  $K$ .

**Initialize:**  $\theta_0 = 0$ , uniform policy  $\pi_0$ .

**For**  $k = 1, 2, \dots, K$ :

1.  $\pi_k(a | x) \propto \pi_{k-1}(a | x) e^{\eta \langle \varphi(x, a), \theta_{k-1} \rangle}$ .
2.  $\hat{g}_k = \tau_E^{-1} \sum_{i=1}^{\tau_E} (\varphi(X_E^i, A_E^i) - \varphi(X_E^i, \pi_k))$ .
3.  $\theta_k = \arg \max_{\theta: \|\theta\| \leq B_\theta} \langle \theta, \hat{g}_k \rangle = \frac{B_\theta}{\|\hat{g}_k\|} \hat{g}_k$ .

**Output:**  $\pi^{\text{out}} = \pi_I$ , where  $I \sim \mathcal{U}([K])$ .

**Algorithm 2** SPOIL with general FA

**Input:** Number of expert trajectories  $\tau_E$ , learning rate  $\eta$ , number of iterations  $K$ .

**Initialize:**  $Q_0 = 0$ , uniform policy  $\pi_0$ .

**For**  $k = 1, 2, \dots, K$ :

1.  $\pi_k(a | x) \propto \pi_{k-1}(a | x) e^{\eta Q_{k-1}(x, a)}$ .
2.  $Q_k \in \arg \max_{Q \in \mathcal{Q}} \hat{\mathcal{L}}(\pi_k, Q)$ .

**Output:**  $\pi^{\text{out}} = \pi_I$ , where  $I \sim \mathcal{U}([K])$ .

**3.1 SPOIL for linear function approximation**

We first provide a set of guarantees under the assumption that the function class is linear in some known features that realize the action-value functions of all policies  $\pi$  as linear combinations (see Assumption 1). In this setting, the actor and critic updates both simplify. For the actor, notice that the policy update can be rewritten as  $\pi_k(a | x) \propto e^{\eta \sum_{i=1}^{k-1} Q_i(x, a)}$ , which only requires storing  $\sum_{i=1}^{k-1} Q_i$  in memory. For linear function approximation, this means that it suffices to maintain a single  $d$ -dimensional vector  $\bar{\theta}_k = \sum_{i=1}^k \theta_i$  in memory and update it incrementally after each critic update. As for the critic update itself, notice that the objective function  $\mathcal{L}$  and its empirical counterpart  $\hat{\mathcal{L}}$  can be rewritten in terms of the gap between the feature-expectation vectors

$$g_k = \mathbb{E}_{(X, A) \sim \mu^{\pi_E}} [\varphi(X, A) - \varphi(X, \pi_k)], \quad \text{and} \quad \hat{g}_k = \frac{1}{\tau_E} \sum_{i=1}^{\tau_E} (\varphi(X_E^i, A_E^i) - \varphi(X_E^i, \pi_k)).$$

When considering linear functions  $Q_\theta(x, a) = \langle \varphi(x, a), \theta \rangle$ , the objective can be written as

$$\mathcal{L}(\pi_k; Q_\theta) = \langle \theta, g_k \rangle, \quad \text{and} \quad \hat{\mathcal{L}}(\pi_k; Q_\theta) = \langle \theta, \hat{g}_k \rangle,$$

and the critic update can be simply written as  $\theta_k = \arg \max_{\theta \in \mathcal{B}(B_\theta)} \langle \theta, \hat{g}_k \rangle$ , which is trivial to compute. All in all, both actor and critic updates can be performed efficiently while only working in a  $d$ -dimensional Euclidean space. The following theorem provides our main result for SPOIL.

**Theorem 2.** *Let Assumption 1 hold. Run Algorithm 1 for  $K = \frac{2 \log |\mathcal{A}|}{(1-\gamma)^2 \varepsilon^2}$  iterations, with a learning rate  $\eta = (1-\gamma) \sqrt{2 \log |\mathcal{A}| / K}$ , and  $\tau_E = \mathcal{O}\left(\frac{d}{(1-\gamma)^2 \varepsilon^2} \log\left(\frac{B_\theta B_\varphi \log |\mathcal{A}|}{(1-\gamma) \varepsilon}\right)\right)$  samples collected by any expert policy  $\pi_E$ . Then, the output satisfies  $\mathbb{E}[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] \leq 5\varepsilon$ .*

The proof is in Appendix B. It is important to highlight that no assumptions are made concerning the expert policy. In particular, we do not require knowledge of a class  $\Pi^E$  realizing the expert policy and as a consequence the bound on  $\tau_E$  does not scale at all with a complexity measure of  $\Pi^E$ . This is in stark contrast with the theoretical guarantees for behavioural cloning (e.g., Agarwal et al., 2019, Chapter 15, and Foster et al., 2024) which show bounds on the expert samples scaling with  $\log |\Pi^E|$  (or the log covering number for continuous classes). It follows that no matter how complex the expert policy is, SPOIL suffers only the complexity of the environment (i.e., the feature dimensionality  $d$ ).

**3.2 SPOIL for general function approximation**

For more complex  $Q^\pi$ -realizable MDPs, we analyze the version of SPOIL given in Algorithm 2. Notice that the updates can no longer use the linear structure of the value functions, and thus the critic update cannot be computed in closed form. Nevertheless, the algorithm remains well-defined, and satisfies the following performance guarantee.

**Theorem 3.** *Let Assumption 2 hold. Run Algorithm 2 for  $K = \frac{2 \log |\mathcal{A}|}{(1-\gamma)^2 \varepsilon^2}$  iterations, with a learning rate  $\eta = (1-\gamma) \sqrt{2 \log |\mathcal{A}| / K}$  and  $\tau_E = \mathcal{O}\left(\frac{\log |\mathcal{A}|}{(1-\gamma)^2 \varepsilon^4} \log\left(\frac{N_{\varepsilon'}(\mathcal{Q}, \|\cdot\|_\infty)}{\varepsilon(1-\gamma)}\right)\right)$  samples collected by any expert policy  $\pi_E$ , where  $\varepsilon' = \frac{(1-\gamma)^3 \varepsilon^2}{4 \log |\mathcal{A}|}$ . Then, the output satisfies  $\mathbb{E}[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] = \mathcal{O}(\varepsilon)$ .*

There are two important remarks for the nonlinear extension. First, the maximization of  $\widehat{\mathcal{L}}(\pi_k, Q)$  with respect to  $Q$  is no longer available in closed form and it might not even be a concave optimization problem depending on the choice of the function class  $\mathcal{Q}$ . Therefore, computational efficiency cannot be ensured. Nevertheless, the form of the objective function remains very simple in terms of  $Q$ , and is arguably easier to optimize than other popular objective functions that are routinely optimized within deep RL with good empirical success (e.g., the objective functions appearing in [Mnih et al., 2015]) and deep IL [Garg et al., 2021]. Secondly, the expert sample complexity bound degrades from  $\mathcal{O}(\varepsilon^{-2})$  achieved in the linear case to  $\mathcal{O}(\varepsilon^{-4})$  in the nonlinear case due to the higher complexity of the policies produced by the algorithm (which results in a larger covering number of the policy class as highlighted in the proof sketch included in the next section).

## 4 Analysis

In this section we outline the proof of our two main results. Both proofs are based on two key steps which are self evident from Proposition 1. The first one consists of a regret analysis to show that  $\sum_{k=1}^K \mathcal{L}(\pi_k, Q_k)$  is bounded sublinearly in  $K$ . At a high level, the proof makes use of a classic technique of decomposing the “global” regret into the average of “local” regrets in each MDP state, first proposed by Even-Dar et al., 2009] and used in numerous other works (e.g., [Abbasi-Yadkori et al., 2019, Geist et al., 2019, Lan, 2023, Moulin and Neu, 2023]). In proving this result, a little care is needed in handling the potentially nonstationary nature of the expert policy. We circumvent the issue by using the performance difference lemma and controlling the regret at each state against the stationary comparator which induces the same state-action occupancy measure of the expert. Formally, we have the following bound, which we prove in Appendix B.

**Lemma 4.** *For any  $k$  and any state-action pair  $(x, a)$ , consider the sequence of policies starting with  $\pi_1$  as the uniform policy and updated as  $\pi_{k+1}(a | x) \propto \pi_k(a | x)e^{\eta Q_k(x, a)}$  for some function  $Q_k : \mathcal{X} \times \mathcal{A} \rightarrow \mathbb{R}$  such that  $\|Q_k\|_\infty \leq \frac{1}{1-\gamma}$ . Then,  $\sum_{k=1}^K \mathcal{L}(\pi_k, Q_k) \leq \frac{\log |\mathcal{A}|}{\eta} + \frac{\eta K}{2(1-\gamma)^2}$ .*

This lemma applies to both the linear and nonlinear settings. The next and final step of the analysis is to establish concentration of the empirical objective and bound  $\Delta(\pi_k)$  for each  $k$ . The main challenge in this step is the correlation between the iterates  $\{\pi_k\}_{k=1}^K$  and the expert dataset. This can be handled via a uniform bound over the policy class to which all the algorithm iterates belong to. Importantly, this class is much smaller than the class of all policies, and allows us to make massive sample-complexity savings as compared to methods that need to control estimation errors associated with arbitrary policies. We provide the technical details separately for the linear and nonlinear cases.

### 4.1 Linear function approximation

In order to bound the estimation errors  $\Delta(\pi_k)$ , we apply a covering argument over the class of linear softmax policies. We have the following result.

**Lemma 5.** *Let  $\{\pi_k\}_{k \in [K]}$  be the sequence of policies generated by Algorithm 1 and let  $\Delta(\pi_k)$  be defined as in Proposition 1. Then, with probability at least  $1 - \delta$ , it holds that*

$$\sum_{k=1}^K \Delta(\pi_k) \leq 1 + 2K \sqrt{\frac{8d}{(1-\gamma)^2 \tau_E} \log \left( \frac{2 + 16B_\theta B_\varphi K}{(1-\gamma)\delta} \right)}.$$

We defer the proof to Appendix B. We can use the above result to sketch the proof of Theorem 2.

*Proof sketch of Theorem 2.* Using Lemma 4 with  $\eta = (1-\gamma)\sqrt{\frac{2\log |\mathcal{A}|}{K}}$  and dividing by  $K$ , we obtain that  $\frac{1}{K} \sum_{k=1}^K \mathcal{L}(\pi_k, Q_k) \leq \sqrt{\frac{2\log |\mathcal{A}|}{(1-\gamma)^2 K}}$ . Therefore, setting  $K = \frac{2\log |\mathcal{A}|}{(1-\gamma)^2 \varepsilon^2}$  guarantees  $\frac{1}{K} \sum_{k=1}^K \mathcal{L}(\pi_k, Q_k) \leq \varepsilon$ . Then, using the high-probability bound in Lemma 5 and the fact that  $K^{-1} \sum_{k=1}^K \Delta(\pi_k)$  is a random variable bounded by  $(1-\gamma)^{-1}$  almost surely, we obtain the following expectation bound which holds for all  $\delta > 0$

$$\frac{1}{K} \sum_{k=1}^K \mathbb{E}[\Delta(\pi_k)] \leq \frac{1}{K} + C \sqrt{\frac{d}{(1-\gamma)^2 \tau_E} \log \left( \frac{B_\theta B_\varphi \log |\mathcal{A}|}{(1-\gamma)^3 \varepsilon^2 \delta} \right)} + \frac{\delta}{1-\gamma},$$

for some  $C \in \mathbb{R}$ . Noticing that the choice of parameters ensures  $\frac{1}{K} \leq \frac{\varepsilon}{2}$  and setting  $\delta = \frac{\varepsilon(1-\gamma)}{2}$  and  $\tau_E \geq \frac{C^2 d}{(1-\gamma)^2 \varepsilon^2} \log\left(\frac{B_\theta B_\varphi \log|\mathcal{A}|}{(1-\gamma)^3 \varepsilon^2 \delta}\right)$ , this bound implies that  $\frac{2}{K} \sum_{k=1}^K \mathbb{E}[\Delta_k] \leq 4\varepsilon$ . Invoking Proposition 1, we conclude that  $\mathbb{E}[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] \leq 5\varepsilon$ . The full proof is in Appendix B.  $\square$

## 4.2 General function approximation

The proof for the nonlinear setup follows the same conceptual steps but requires a more general concentration result for the objective function. Namely, the following lemma is the general counterpart of Lemma 5. The feature dimension  $d$  appearing in the linear case is replaced by the complexity (as measured by the covering number) of the policy and value function classes containing the iterates.

**Lemma 6.** *Let  $\Pi_Q$  denote the policy class containing the iterates  $\{\pi_k\}_{k=1}^K$  produced by Algorithm 2, then with probability at least  $1 - \delta$ , for all  $k \in [K]$  it holds that*

$$\Delta(\pi_k) = \sup_{Q \in \mathcal{Q}} |\hat{\mathcal{L}}(\pi_k, Q) - \mathcal{L}(\pi_k, Q)| \leq \frac{1}{2K} + \sqrt{\frac{2(K+1) \log(2\mathcal{N}_{(1-\gamma)/8K}(\mathcal{Q}, \|\cdot\|_\infty)/\delta)}{(1-\gamma)^2 \tau_E}}.$$

The proof is in Appendix B. Note that in the nonlinear case, the complexity of the policy class increases linearly with the number of iterations  $K$  (see Lemma 12). On the contrary, in the linear case, the policies generated by Algorithm 1 are parameterized by  $d$  parameters and only the magnitude of these parameters increases with  $K$ . With this lemma, we present the proof sketch of Theorem 3.

*Proof sketch of Theorem 3.* Applying the decomposition in Proposition 1, the regret bound in Lemma 4, the concentration in Lemma 6, we obtain  $\mathbb{E}[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] = \tilde{O}\left(\frac{1}{\sqrt{K}} + \sqrt{\frac{K}{\tau_E}}\right)$ . Setting  $K = \tilde{O}(\varepsilon^{-2})$ , and  $\tau_E = \tilde{O}(\varepsilon^{-4})$ , we get  $\mathbb{E}[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] = \varepsilon$ . The full proof is in Appendix B.  $\square$

## 5 Numerical experiments

We run experiments to verify that we can imitate efficiently complex experts in linear  $Q^\pi$  environment, and may achieve massive improvements over behavioral cloning with large policy classes.

To investigate this, we consider a randomly generated large linear MDP (a particular case of linear  $Q^\pi$ -realizable MDP) with  $|\mathcal{X}| = 500$  and  $|\mathcal{A}| = 1000$  but with a small feature dimension  $d = 7$ . We instantiate two experts. A first expert is trained to be the soft optimal policy in this environment which is parametrized by only  $d$  parameters and it can be realized by the following policy class  $\Pi_{\text{lin}}^E = \left\{ \pi(a|x) = \frac{\exp(\langle \varphi(x, a), \theta \rangle)}{\sum_{b \in \mathcal{A}} \exp(\langle \varphi(x, b), \theta \rangle)}, \theta \in \mathbb{R}^d, \|\theta\| \leq B_\theta \right\}$ . In addition, we consider a second expert belonging to the class of three-layer neural networks denoted by  $\Pi_{\text{NN}}^E$ . This expert was trained to minimize the KL divergence with respect to the linear expert. As evident from Figure 1, our algorithm SPOIL performs well for both experts. This is in perfect agreement with the theory which provides a sample complexity bound which is independent of the expert policy class. On the other hand, behavioural cloning (BC) struggles with the complexity of neural network expert policy class, and performs much worse. This is despite the fact that the data sets perfectly satisfy the realizability condition required by BC. This clearly demonstrates that complex behavior policies may indeed be problematic for BC to deal with, and we expect that such issues may cause real performance drops in practical applications as well. Notice that in this experiments SPOIL outperforms BC because the environment complexity is much lower than the

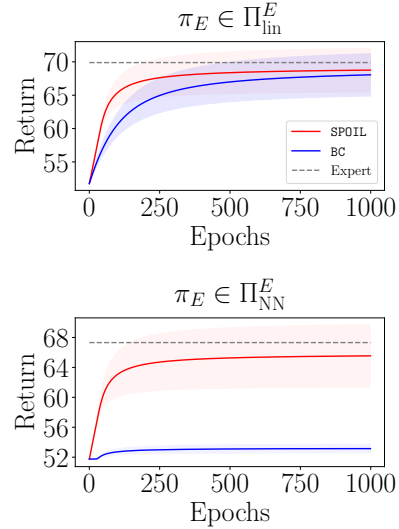


Figure 1: Experiments with simple and complex experts. Curves are averaged across 10 seeds.



314 policy class. For fairness, we point out that the opposite situation is not unusual in RL and IL. In that  
 315 case, it is reasonable to expect BC to be superior to SPOIL.

## 316 5.1 Continuous states experiments

317 We run the general function approximation version of our algorithm in continuous states environments  
 318 from the gym library. In particular, we consider the environments `CartPole-v1`, `Acrobot-v1`  
 319 and `LunarLander-v2` where the expert is trained via `Soft DQN`. In particular, we use the expert  
 320 data provided in the code base of [Garg et al. \[2021\]](#). The learner aims at imitating the expert  
 321 performance given as input a variable number of expert trajectories. In order to make the task  
 322 more challenging the trajectories are subsampled each 20 steps in `CartPole-v1`, `Acrobot-v1` and  
 323 each 5 in `LunarLander-v2`.<sup>1</sup> We compare the performance of the best policy found by each of  
 these algorithms as a function of the number of expert trajectories given as input. In practice the

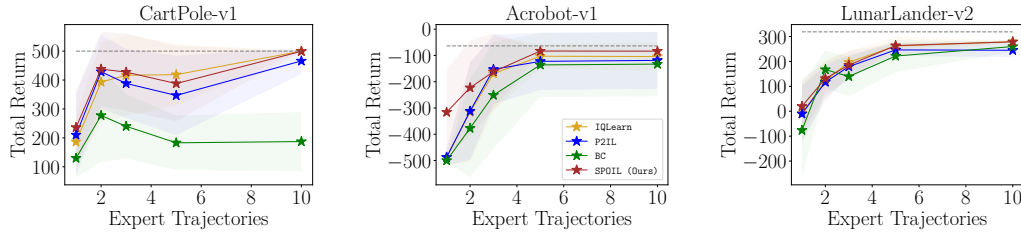


Figure 2: Experiments in continuous states domains. Curves are averaged across 10 seeds.

324 maximization  $\arg \max_{Q \in \mathcal{Q}} \hat{\mathcal{L}}(\pi_k, Q)$  is performed only approximately by performing a gradient  
 325 ascent step. On the other hand, the actor update is approximated via `Soft DQN` [Haarnoja et al. \[2017\]](#).  
 326 In Figure 2, we can see that SPOIL compares comparably to the state of the art algorithm IQ-Learn  
 327 [Garg et al. \[2021\]](#) and improves upon BC [Pomerleau \[1991\]](#), [Foster et al. \[2024\]](#) and P<sup>2</sup>IL [Viano et al. \[2022\]](#).  
 328  
 329

## 330 6 Conclusions

331 In this work, we proposed analyses that leverages structural assumptions on the MDP without  
 332 requiring trajectory access. This is made possible thanks to a novel regret decomposition that shifts  
 333 the focus from updating a reward sequence based on expert data to updating a sequence of state-action  
 334 value functions. To the best of our knowledge, these are the first rigorous theoretical guarantees for IL  
 335 methods that learn state-action value functions from expert data, a technique popularized in practice  
 336 by [Garg et al. \[2021\]](#). Among the many potential ways to improve extend and improve our work, we  
 337 highlight a handful of directions in Appendix F.

338 All previous theory work has focused either on imitation learning with additional trajectory access  
 339 to the environment, both in tabular MDPs [[Shani et al., 2021](#), [Xu et al., 2023](#)] and with additional  
 340 structural assumptions [[Liu et al., 2022](#), [Viano et al., 2022, 2024](#), [Moulin et al., 2025](#)], or learning  
 341 based on offline data only but under structural assumptions about the policy class used by the expert  
 342 [[Rajaraman et al., 2021](#), [Swamy et al., 2022](#), [Foster et al., 2024](#), [Rohatgi et al., 2025](#)]. The first of  
 343 these assumptions is clearly more restrictive than what we have considered in this work, and we have  
 344 pointed out potential issues with the second set of methods when the policy class is exceedingly  
 345 complex. This is not to say though that we consider our approach strictly superior to policy-based  
 346 IL methods: as is often the case in RL, there is no single approach that dominates all others in all  
 347 problems, and sometimes policy-based methods are more suitable for the job than value-based ones.  
 348 Thus, even if our approach is not the ultimate answer to all questions in imitation learning, our results  
 349 show that it is one potential alternative to consider in situations where other methods fail.

<sup>1</sup>This is common practice in IL experiments (see, e.g., [Garg et al., 2021](#)).

## References

- Yasin Abbasi-Yadkori, Peter Bartlett, Kush Bhatia, Nevena Lazic, Csaba Szepesvari, and Gellért Weisz. PoliteX: Regret bounds for policy iteration using expert prediction. In *International Conference on Machine Learning (ICML)*, 2019.
- P. Abbeel and A. Y. Ng. Apprenticeship learning via inverse reinforcement learning. In *International Conference on Machine Learning (ICML)*, 2004.
- Alekh Agarwal, Nan Jiang, Sham M Kakade, and Wen Sun. Reinforcement learning: Theory and algorithms. *CS Dept., UW Seattle, Seattle, WA, USA, Tech. Rep.*, 32, 2019.
- Adam Block, Ali Jadbabaie, Daniel Pfrommer, Max Simchowitz, and Russ Tedrake. Provable guarantees for generative behavior cloning: Bridging low-level stability and high-level behavior. *Advances in Neural Information Processing Systems*, 36:48534–48547, 2023.
- Nicolo Cesa-Bianchi and Gábor Lugosi. *Prediction, learning, and games*. Cambridge university press, 2006.
- Nicolas Espinosa-Dice, Sanjiban Choudhury, Wen Sun, and Gokul Swamy. Efficient imitation under misspecification. *arXiv preprint arXiv:2503.13162*, 2025.
- E. Even-Dar, S. M. Kakade, and Y. Mansour. Experts in a Markov decision process. In *Neural Information Processing Systems*, pages 401–408.
- Eyal Even-Dar, Sham. M. Kakade, and Yishay Mansour. Online Markov decision processes. *Mathematics of Operations Research*, 34(3):726–736, 2009.
- Dylan J Foster, Adam Block, and Dipendra Misra. Is behavior cloning all you need? understanding horizon in imitation learning. *arXiv preprint arXiv:2407.15007*, 2024.
- Bolin Gao and Laca Pavel. On the properties of the softmax function with application in game theory and reinforcement learning. *arXiv preprint arXiv:1704.00805*, 2017.
- Divyansh Garg, Shuvam Chakraborty, Chris Cundy, Jiaming Song, and Stefano Ermon. IQ-learn: Inverse soft-Q learning for imitation. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.
- Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A Theory of Regularized Markov Decision Processes. In *International Conference on Machine Learning (ICML)*, 2019.
- Tuomas Haarnoja, Haoran Tang, Pieter Abbeel, and Sergey Levine. Reinforcement learning with deep energy-based policies. In *Proceedings of the 34th International Conference on Machine Learning*, pages 1352–1361. PMLR, 2017.
- Ronald A Howard. Dynamic programming and Markov processes. 1960.
- Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *Proceedings of the nineteenth international conference on machine learning*, pages 267–274, 2002.
- Diederik P Kingma and Jimmy Ba. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.
- Guanghui Lan. Policy mirror descent for reinforcement learning: Linear convergence, new sampling complexity, and generalized problem classes. *Mathematical programming*, 198(1):1059–1106, 2023.
- Zhihan Liu, Yufeng Zhang, Zuyue Fu, Zhuoran Yang, and Zhaoran Wang. Learning from demonstration: Provably efficient adversarial policy imitation with linear function approximation. In *International Conference on Machine Learning (ICML)*, 2022.
- Zakaria Mhammedi. Sample and oracle efficient reinforcement learning for mdps with linearly-realizable value functions. *arXiv preprint arXiv:2409.04840*, 2024.

Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A. Rusu, Joel Veness, Marc G. Belle-  
mare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen,  
Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dhharshan Kumaran, Daan Wierstra,  
Shane Legg, and Demis Hassabis. Human-level control through deep reinforcement learning.  
*Nature*, 518(7540):529–533, 2015.

Antoine Moulin and Gergely Neu. Optimistic planning by regularized dynamic programming.  
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 25337–25357. PMLR, 23–29  
Jul 2023. URL <https://proceedings.mlr.press/v202/moulin23a.html>.

Antoine Moulin, Gergely Neu, and Luca Viano. Optimistically optimistic exploration for provably  
efficient infinite-horizon reinforcement and imitation learning. *arXiv preprint arXiv:2502.13900*,  
2025.

Francesco Orabona. A modern introduction to online learning, 2023.

A Paszke. Pytorch: An imperative style, high-performance deep learning library. *arXiv preprint  
arXiv:1912.01703*, 2019.

D. A. Pomerleau. Efficient training of artificial neural networks for autonomous navigation. *Neural  
Computation*, 3(1):88–97, 1991.

M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John  
Wiley & Sons, Inc., USA, 1st edition, 1994.

Nived Rajaraman, Lin Yang, Jiantao Jiao, and Kannan Ramchandran. Toward the fundamental limits  
of imitation learning. *Advances in Neural Information Processing Systems*, 33:2914–2924, 2020.

Nived Rajaraman, Yanjun Han, Lin Yang, Jingbo Liu, Jiantao Jiao, and Kannan Ramchandran. On  
the value of interaction and function approximation in imitation learning. *Advances in Neural  
Information Processing Systems*, 34:1325–1336, 2021.

Dhruv Rohatgi, Adam Block, Audrey Huang, Akshay Krishnamurthy, and Dylan J Foster.  
Computational-statistical tradeoffs at the next-token prediction barrier: Autoregressive and imita-  
tion learning under misspecification. *arXiv preprint arXiv:2502.12465*, 2025.

S. Ross, G. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction  
to no-regret online learning. In *International Conference on Artificial Intelligence and Statistics  
(AISTATS)*, 2011.

Stéphane Ross and Drew Bagnell. Efficient reductions for imitation learning. In *International  
Conference on Artificial Intelligence and Statistics (AISTATS)*, 2010.

Lior Shani, Tom Zahavy, and Shie Mannor. Online apprenticeship learning. *arXiv:2102.06924*, 2021.

Max Simchowitz, Daniel Pfrommer, and Ali Jadbabaie. The pitfalls of imitation learning when  
actions are continuous. *arXiv preprint arXiv:2503.09722*, 2025.

Gokul Swamy, Sanjiban Choudhury, J Andrew Bagnell, and Steven Wu. Of moments and matching:  
A game-theoretic framework for closing the imitation gap. In *International Conference on Machine  
Learning*, pages 10022–10032. PMLR, 2021.

Gokul Swamy, Nived Rajaraman, Matt Peng, Sanjiban Choudhury, J Bagnell, Steven Z Wu, Jiantao  
Jiao, and Kannan Ramchandran. Minimax optimal online imitation learning via replay estimation.  
*Advances in Neural Information Processing Systems*, 35:7077–7088, 2022.

U. Syed and R. E. Schapire. A game-theoretic approach to apprenticeship learning. In *Advances in  
Neural Information Processing Systems (NeurIPS)*, 2007.

Volodymyr Tkachuk, Gellért Weisz, and Csaba Szepesvari. Trajectory data suffices for statistically  
efficient learning in offline RL with linear  $q^\pi$ -realizability and concentrability. In *The  
Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL <https://openreview.net/forum?id=TusuJSbRxm>.

- 442 Luca Viano, Angeliki Kamoutsis, Gergely Neu, Igor Krawczuk, and Volkan Cevher. Proximal point  
443 imitation learning. *Advances in Neural Information Processing Systems*, 35:24309–24326, 2022.
- 444 Luca Viano, Stratis Skoulakis, and Volkan Cevher. Imitation learning in discounted linear MDPs  
445 without exploration assumptions. In *Forty-first International Conference on Machine Learning*,  
446 2024. URL <https://openreview.net/forum?id=DChQpB4AJy>.
- 447 Gellért Weisz, András György, and Csaba Szepesvári. Online rl in linearly  $q^\pi$ -realizable mdps is as  
448 easy as in linear mdps if you learn what to ignore. *Advances in Neural Information Processing*  
449 *Systems*, 36:59172–59205, 2023.
- 450 Tian Xu, Ziniu Li, Yang Yu, and Zhi-Quan Luo. Provably efficient adversarial imitation learning  
451 with unknown transitions. *arXiv preprint arXiv:2306.06563*, 2023.

## Contents of Appendix

<b>A Additional related works</b>	<b>13</b>
<b>B Omitted proofs</b>	<b>15</b>
B.1 Proof of Lemma 1 (performance difference lemma)	15
B.2 Proof of Lemma 4 (regret of the policy player)	15
B.3 General concentration argument	16
B.4 Proof of Lemma 5 (concentration linear case)	18
B.5 Proof of Theorem 2 (sample complexity guarantee for linear $Q^\pi$ -realizable MDPs)	20
B.6 Proof of Lemma 6 (concentration nonlinear case)	20
B.7 Proof of Theorem 3 (sample complexity guarantee for $Q^\pi$ -realizable MDPs)	22
<b>C Technical tools</b>	<b>23</b>
<b>D On the guarantees of misspecified BC in linear <math>Q^\pi</math>-realizable MDPs</b>	<b>24</b>
<b>E Experimental details</b>	<b>24</b>
<b>F Future directions</b>	<b>25</b>

## A Additional related works

Classical analyses by [Ross and Bagnell \[2010\]](#), [Ross et al. \[2011\]](#) on behavioural cloning (BC) established an error propagation framework relating the suboptimality of the learned policy to the worst-case generalization error incurred in predicting the expert policy. They proved that this suboptimality gap is upper-bounded by the generalization error up to a multiplicative factor  $H^2$  (where  $H$  is the horizon), a factor that is unavoidable when using the 0-1 loss for supervised learning. However, these results do not quantify the *expert sample complexity*, or the number of samples required to make the generalization error small.

A recent line of work has begun to investigate the expert sample complexity assuming knowledge of a policy class  $\Pi^E$  that realizes (or nearly realizes) the expert policy. For instance, [Rajaraman et al. \[2021\]](#) assume that the expert is deterministic and belongs to the class of deterministic linear policies  $\Pi_{\text{det}, \text{lin}}$  (defined in the caption of Table 1). They prove a bound on the required number of expert samples of order  $\tilde{O}((H^2 d) / \varepsilon)$ , where  $d$  is the feature dimension in the definition of  $\Pi_{\text{det}, \text{lin}}$ . Their technique is a reduction to the problem of multiclass classification in supervised learning, but their result is not informative for settings with general stochastic expert policies.

Further contributions to understanding the sample complexity of IL under policy class assumptions were made by [Foster et al. \[2024\]](#). Specifically, assuming the expert is included within a known class,  $\pi_E \in \Pi^E$ , they showed that one can learn an  $\varepsilon$ -optimal policy (as defined in Equation (2)) after observing  $\mathcal{O}((H^2 \log |\Pi^E|) / \varepsilon)$  samples for a deterministic expert or  $\mathcal{O}((H^2 \log |\Pi^E|) / \varepsilon^2)$  samples for a stochastic one (we report the dense reward case for brevity, though their bounds improve for sparse rewards). Addressing scenarios where the expert policy might only be almost well-specified, [Rohatgi et al. \[2025\]](#) demonstrate that there exists a computationally efficient algorithm that outputs an  $\varepsilon$ -optimal policy up to an additional approximation error of  $H \log(W) \min_{\pi \in \Pi^E} \mathcal{D}_H^2(\mathbb{P}^\pi, \mathbb{P}^{\pi_E})$ . In this context,  $\mathbb{P}^\pi$  is the trajectory distribution induced by  $\pi$ ,  $W$  is a density ratio defined as

$$W = \max_{\pi \in \Pi^E} \max_{(x, a) \in \mathcal{X} \times \mathcal{A}} \max_{h \in [H]} \frac{\pi_{E, h}(a | x)}{\pi_h(a | x)}.$$

It is worth noting that these guarantees become vacuous when the policy class  $\Pi^E$  is such that at least one policy in  $\Pi^E$  fails to provide sufficient coverage for the expert’s actions (leading to  $W = +\infty$  as  $\pi_h(a | x)$  gets close to zero for relevant state-action pairs and timestep where  $\pi_{E, h}(a | x) > 0$ ), or if the minimum Hellinger distance  $\min_{\pi \in \Pi^E} \mathcal{D}_H^2(\mathbb{P}^\pi, \mathbb{P}^{\pi_E})$  is large. Alternatively, [Foster et al. \[2024\]](#) proved a misspecification result where the additional error is  $\min_{\pi \in \Pi^E} \chi^2(\mathbb{P}^\pi, \mathbb{P}^{\pi_E})$ . This misspecification error is measured by the  $\chi^2$  divergence, with a leading coefficient constant in  $H$  and



496  $W$ . However, the  $\chi^2$  divergence is an upper bound on the Hellinger distance that is often way too  
 497 loose to be practical. In a similar vein, [Espinosa-Dice et al. \[2025\]](#) proved a benefit in terms of error  
 498 propagation for a local search algorithm over behavioural cloning in misspecified settings, under the  
 499 assumption that the learned policy is allowed to reset to states visited in the expert dataset.

500 Our work aligns with the recent renewed interest in proving refined expert sample complexity  
 501 guarantees for offline imitation learning but distinguishes itself by swapping out the expert realizability  
 502 assumption with a structural assumption on the environment. Early explorations for similar settings  
 503 can be found in classical works by [Abbeel and Ng \[2004\]](#) and [Syed and Schapire \[2007\]](#). These  
 504 studies proposed offline learning algorithms for MDPs with reward functions linear in a collection of  
 505 features known to the learner, under the assumption that transition dynamics of the environment is  
 506 also known. Versions of their approaches that do not assume such knowledge typically incur a worse  
 507 sample complexity and often apply only in the tabular setting. Our work generalizes these classical  
 508 approaches by removing the need for known transitions and for rewards to be linear in the features, as  
 509 well as going beyond tabular MDPs. Notably, the linear  $Q^\pi$ -realizability assumption can hold even if  
 510 the reward function and the transition dynamics are nonlinear. We summarize our comparison with  
 511 these and other related works in Table 1.

512 Our work focuses on *learning a  $Q$ -value from expert data* and, in this regard, is closely related to the  
 513 practical work of [Garg et al. \[2021\]](#). The novel regret decomposition employed in our analysis of  
 514 SPOIL demonstrates, we believe for the first time, that provable guarantees are achievable by directly  
 515 learning an action-value function from expert data. This contrasts with the majority of theoretical and  
 516 practical imitation learning approaches, which typically first use the expert data to learn a reward  
 517 function and subsequently use this learned reward function to infer an action-value function.

518 Moreover, we note that SPOIL shares similarities with the algorithm AdvIL proposed by [Swamy et al.](#)  
 519 [\[2021\]](#). Specifically, both SPOIL and AdvIL consider the same objective  $\mathcal{L}$  but the two methods differ  
 520 in their proposed algorithmic solutions and analytical approaches. Notably, [Swamy et al. \[2021\]](#)  
 521 employed simultaneous gradient descent-ascent updates that made little use of the specific problem  
 522 structure, whereas we consider an asymmetric scheme where the policy player uses mirror descent  
 523 and the  $Q$ -player plays the best response. As mentioned previously in relation to the work of [Garg](#)  
 524 [et al. \[2021\]](#), our approach is more akin to minimizing the function  $\pi \mapsto \max_{Q \in \mathcal{Q}} \mathcal{L}(\pi, Q)$  rather  
 525 than using a primal-dual scheme.

526 A key difference lies in the analysis: [Swamy et al. \[2021\]](#) conduct an error propagation analysis  
 527 for AdvIL. From this, they conclude that AdvIL is equivalent to BC in the sense that if the loss for  
 528 either method is at most  $\varepsilon$  in every state, then the suboptimality of the extracted policy in an episodic  
 529 setting with horizon  $H$  is of order  $H^2\varepsilon$  for both. However, this type of result does not investigate the  
 530 assumptions or the number of samples needed to ensure these losses are indeed less than  $\varepsilon$ . Our work  
 531 addresses this open question, establishing a clear distinction between the sample complexities of  
 532 SPOIL and BC. Specifically, SPOIL and BC (and their respective analyses) rely on largely orthogonal  
 533 sets of assumptions, making the two approaches complementary to each other: we expect SPOIL to  
 534 be more suitable for imitation tasks with complex experts but simpler environments, while BC may  
 535 be the preferred choice when this situation is reversed. Our sample complexity analysis for SPOIL  
 536 critically relies on the  $Q$ -player using a best response strategy, and it is unlikely that equivalent results  
 537 could be achieved using a standard gradient ascent step for the  $Q$ -player instead.

538 Very recently, [Simchowitz et al. \[2025\]](#) analyzed the error propagation properties of offline imitation  
 539 learning algorithms in continuous action MDPs, showing that an exponential dependence on the  
 540 horizon of the problem is unavoidable if no structure is imposed on the environment. On the other  
 541 hand, the same authors point out that if the state-action value functions were Lipschitz in the action  
 542 space, then efficient learning would be possible. Conceptually, we believe that the SPOIL algorithm  
 543 could also be applied in the continuous action case. Such an extension would suggest that another  
 544 scenario enabling effective imitation learning in continuous action spaces arises when the learner has  
 545 access to a suitably expressive class of state-action value functions.

546 Following a similar line of research that studies imitation learning from a control-theoretic perspective,  
 547 [Block et al. \[2023\]](#) studied guarantees for generative behavioural cloning, assuming access to a  
 548 stabilizing policy dubbed a *synthesis oracle*. These policies can be computed exactly if the dynamics  
 549 are known, an assumption which is not imposed in our work. However, when provided with such  
 550 an oracle, [Block et al. \[2023\]](#) derive bounds on a stricter metric for imitation. Specifically, they

bound the probability that expert and learner trajectories diverge at some time step, as opposed to the difference in cumulative return that we analyze in our work.

## B Omitted proofs

In this appendix, we provide the omitted proofs of the main results.

### B.1 Proof of Lemma 1 (performance difference lemma)

We start presenting the performance difference lemma proven in a more general form which allows one policy to be nonstationary.

**Lemma 7.** *Let  $\pi$  be a stationary policy and  $\pi'$  be any policy. Then,*

$$\rho^{\pi'} - \rho^\pi = \mathbb{E}_{(X,A) \sim \mu^{\pi'}} [Q^\pi(X, A) - V^\pi(X)].$$

*Proof.* Consider the Bellman equations for the stationary policy  $\pi$ . For any state-action pair  $(x, a)$ , we have

$$Q^\pi(x, a) = r(x, a) + \gamma \sum_{x' \in \mathcal{X}} P(x' | x, a) V^\pi(x').$$

Averaging both sides with the distribution  $\mu^{\pi'}$  and reordering the terms, we obtain

$$\begin{aligned} \sum_{x,a} \mu^{\pi'}(x, a) r(x, a) &= \sum_{x,a} \mu^{\pi'}(x, a) \left( Q^\pi(x, a) - \gamma \sum_{x' \in \mathcal{X}} P(x' | x, a) V^\pi(x') \right) \\ &= (1 - \gamma) \sum_x \nu_0(x) V^\pi(x) + \sum_{x,a} \mu^{\pi'}(x, a) (Q^\pi(x, a) - V^\pi(x)), \end{aligned}$$

where we used the flow condition of the occupancy measure  $\mu^{\pi'}$  in the last step (see Equation 1). The claim then follows by noticing that  $\rho^\pi = (1 - \gamma) \sum_x \nu_0(x) V^\pi(x)$  and  $\rho^{\pi'} = \sum_{x,a} \mu^{\pi'}(x, a) r(x, a)$ .  $\square$

### B.2 Proof of Lemma 4 (regret of the policy player)

Next, we apply Lemma 15 to the special case of the exponential weights update, where the divergence is chosen to be the KL divergence, and use it to derive a bound on the regret of the policy player.

**Lemma 8.** *For any  $k$  and any state-action pair  $(x, a)$ , consider the sequence of policies starting with  $\pi_1$  as the uniform policy and updated as  $\pi_{k+1}(a | x) \propto \pi_k(a | x) e^{\eta Q_k(x, a)}$  for some function  $Q_k: \mathcal{X} \times \mathcal{A} \rightarrow \mathbb{R}$  such that  $\|Q_k\|_\infty \leq \frac{1}{1-\gamma}$ . Then,  $\sum_{k=1}^K \mathcal{L}(\pi_k, Q_k) \leq \frac{\log|\mathcal{A}|}{\eta} + \frac{\eta K}{2(1-\gamma)^2}$ .*

*Proof.* Let us recall that

$$\mathcal{L}(\pi_k, Q_k) = \mathbb{E}_{(X,A) \sim \mu^{\pi_k}} [Q_k(X, A) - Q_k(X, \pi_k)],$$

where  $\pi_k$  is a potentially nonstationary policy. To continue, let us consider the stationary policy  $\bar{\pi}_k: \mathcal{X} \rightarrow \Delta(\mathcal{A})$  that induces the same state-action occupancy measure of the expert, i.e., such that  $\mu^{\bar{\pi}_k} = \mu^{\pi_k}$ . This equality can be guaranteed by choosing, for any  $(x, a) \in \mathcal{X} \times \mathcal{A}$ ,  $\bar{\pi}_k(a | x) = \frac{\mu^{\pi_k}(x, a)}{\nu^{\pi_k}(x)}$  if  $\nu^{\pi_k}(x) \neq 0$  and  $\pi_0(a)$  otherwise, where  $\pi_0 \in \Delta(\mathcal{A})$  is an arbitrary distribution. Then, we continue as follows

$$\begin{aligned} \mathcal{L}(\pi_k, Q_k) &= \mathbb{E}_{(X,A) \sim \mu^{\pi_k}} [Q_k(X, A) - Q_k(X, \pi_k)] \\ &= \mathbb{E}_{(X,A) \sim \mu^{\bar{\pi}_k}} [Q_k(X, A) - Q_k(X, \pi_k)] \\ &= \sum_{x \in \mathcal{X}} \nu^{\bar{\pi}_k}(x) \sum_{a \in \mathcal{A}} Q_k(x, a) (\bar{\pi}_k(a | x) - \pi_k(a | x)). \end{aligned}$$

Summing over  $k \in [K]$ , we obtain

$$\sum_{k=1}^K \mathcal{L}(\pi_k, Q_k) = \sum_{x \in \mathcal{X}} \nu^{\bar{\pi}_k}(x) \sum_{k=1}^K \sum_{a \in \mathcal{A}} Q_k(x, a) (\bar{\pi}_k(a | x) - \pi_k(a | x)).$$

578 It remains to prove the following bound.

$$\sum_{k=1}^K \sum_{a \in \mathcal{A}} Q_k(x, a) (\bar{\pi}_E(a | x) - \pi_k(a | x)) \leq \frac{\log |\mathcal{A}|}{\eta} + \frac{\eta K}{2(1 - \gamma)^2}.$$

579 The result is proven as a particular case of Lemma 15. Specifically, we have that when  $V$  is the  
580  $|\mathcal{A}|$ -dimensional simplex and the Bregman divergence is the KL divergence, it holds that

$$x_{k+1} = \arg \min_{v \in V} \left\{ \langle \ell_k, v \rangle + \frac{1}{\eta} D(v, x_k) \right\} = \frac{x_k \odot \exp(-\eta \ell_k)}{\langle \mathbf{1}, x_k \odot \exp(-\eta \ell_k) \rangle},$$

581 where  $\odot$  is the elementwise product. We apply Lemma 15 for each state  $x \in \mathcal{X}$ , replacing  
582  $x_k = \pi_k(\cdot | x)$  and  $\ell_k = -Q_k(x, \cdot)$ . We obtain that for the update  $\pi_{k+1}(a | x) \propto \pi_k(a | x) e^{\eta Q_k(x, a)}$ ,  
583 the guarantee in Lemma 15 holds. Moreover, in this setting we have  $\lambda = 1$ , and  $\ell_{\max} = \frac{1}{1 - \gamma}$ .  
584 Given that for any state-action pair  $(x, a)$ , the initial policy is  $\pi_1(a | x) = \frac{1}{|\mathcal{A}|}$ , we have that  
585  $D(\pi(\cdot | x), \pi_1(\cdot | x)) \leq \log |\mathcal{A}|$ . Thus, we have the following bound

$$\sum_{a \in \mathcal{A}} Q_k(x, a) (\bar{\pi}_E(a | x) - \pi_k(a | x)) \leq \frac{\log |\mathcal{A}|}{\eta} + \frac{\eta K}{2(1 - \gamma)^2},$$

586 and the conclusion follows from  $\nu^{\bar{\pi}_E}$  being a probability distribution.  $\square$

### 587 B.3 General concentration argument

588 To prove the main results of this paper, we prove a general concentration inequality that we will  
589 use for the iterates produced by both Algorithm 1 and Algorithm 2. Specifically, when analyzing  
590 Algorithm 1, we consider the policy class  $\Pi_{\text{lin}}$  defined as follows

$$\Pi_{\text{lin}} = \left\{ \pi \in \Delta(\mathcal{A})^{\mathcal{X}} : \exists (\theta_k)_{k \in [K]} \subset \mathfrak{B}(B_\theta), \pi(a | x) = \frac{\exp\left(\eta \sum_{k=1}^K \langle \varphi(x, a), \theta_k \rangle\right)}{\sum_{b \in \mathcal{A}} \exp\left(\eta \sum_{k=1}^K \langle \varphi(x, b), \theta_k \rangle\right)} \right\}, \quad (3)$$

591 while in the nonlinear case (Algorithm 2), we will consider the policy class

$$\Pi_{\mathcal{Q}} = \left\{ \pi \in \Delta(\mathcal{A})^{\mathcal{X}} : \exists (Q_k)_{k \in [K]} \subset \mathcal{Q}, \pi(a | x) = \frac{\exp\left(\eta \sum_{k=1}^K Q_k(x, a)\right)}{\sum_{b \in \mathcal{A}} \exp\left(\eta \sum_{k=1}^K Q_k(x, b)\right)} \right\}. \quad (4)$$

592 The result is the following.

593 **Lemma 9.** *Let us consider a value function class  $\mathcal{Q} \subset \mathbb{R}^{\mathcal{X} \times \mathcal{A}}$  and the sequences of estimated  
594 objective functions  $\{\hat{\mathcal{L}}(\pi_k, Q)\}_{k=1}^K$  for a policy sequence  $\{\pi_k\}_{k=1}^K$  belonging to a policy class  $\Pi$ .  
595 For any  $k \in [K]$ , recall that for any policy  $\pi$  and function  $Q$ , the objective function is defined as*

$$\mathcal{L}(\pi; Q) = \mathbb{E}_{(X, A) \sim \mu^{\pi_E}} [Q(X, A) - Q(X, \pi)].$$

596 Then, with probability larger than  $1 - \delta$ , it holds that for all  $k \in [K]$  simultaneously that

$$\Delta(\pi_k) = \sup_{Q \in \mathcal{Q}} \left| \hat{\mathcal{L}}(\pi_k, Q) - \mathcal{L}(\pi_k, Q) \right| \leq \inf_{\epsilon: \epsilon > 0} \left\{ \frac{4\epsilon}{1 - \gamma} + \sqrt{\frac{2 \log\left(2\mathcal{N}_\epsilon\left(\mathcal{Q} \times \Pi, \|\cdot\|_{\infty, 1}\right)/\delta\right)}{(1 - \gamma)^2 \tau_E}} \right\},$$

597 where, for any  $(Q, \pi) \in \mathcal{Q} \times \Pi$ , we defined the norm  $\|(Q, \pi)\|_{\infty, 1} = \|Q\|_\infty + \max_{x \in \mathcal{X}} \|\pi(\cdot | x)\|_1$ .

598 *Proof.* Let us recall that for any  $Q \in \mathcal{Q}$  and any  $k \in [K]$ , we have

$$\hat{\mathcal{L}}(\pi_k, Q) = \frac{1}{\tau_E} \sum_{i=1}^{\tau_E} \left( Q(X_E^i, A_E^i) - \sum_{a \in \mathcal{A}} \pi_k(a | X_E^i) Q(X_E^i, a) \right),$$

599 and notice that  $\widehat{\mathcal{L}}(\pi_k, Q)$  is not an unbiased estimator of  $\mathcal{L}(\pi_k, Q)$  since the policy  $\pi_k$  depends on the  
600 expert data. Therefore, we aim at establishing a uniform concentration bound over the policy class  $\Pi$ .  
601 To this end, let us consider a fixed pair  $(Q, \pi) \in \mathcal{C}_\epsilon(\mathcal{Q} \times \Pi, \|\cdot\|_{\infty,1})$ , and notice that  $\widehat{\mathcal{L}}(\pi, Q)$  is a  
602 sum of random variables of the form

$$W_i = \frac{1}{\tau_E} \left( Q(X_E^i, A_E^i) - \sum_{a \in \mathcal{A}} \pi(a | X_E^i) Q(X_E^i, a) \right),$$

603 where  $i \in [\tau_E]$ . Each  $W_i$  is an unbiased estimator of  $\mathcal{L}(\pi, Q)$  since  $\pi$  is fixed (*i.e.*,  $\pi$  is not a random  
604 quantity depending on the expert data) and  $(X_E^i, A_E^i) \sim \mu^{\pi_E}$  for all  $i \in [\tau_E]$ . Thus, for any  $i \in [\tau_E]$ ,  
605  $\mathbb{E}[W_i] = \mathcal{L}(\pi, Q)$ . Moreover, notice that for all  $i \in [\tau_E]$ ,  $-\frac{1}{\tau_E(1-\gamma)} \leq W_i \leq \frac{1}{\tau_E(1-\gamma)}$ . Therefore,  
606 by an application of Hoeffding's inequality (see Lemma 14), we have that for all  $t > 0$ ,

$$\mathbb{P} \left[ \left| \widehat{\mathcal{L}}(\pi, Q) - \mathcal{L}(\pi, Q) \right| \geq t \right] \leq 2 \exp \left( -\frac{2t^2 \tau_E (1-\gamma)^2}{4} \right).$$

607 That is, choosing  $t = \frac{2 \log(2/\delta)}{(1-\gamma)^2 \tau_E}$  guarantees that with probability at least  $1 - \delta$ ,

$$\left| \widehat{\mathcal{L}}(\pi, Q) - \mathcal{L}(\pi, Q) \right| \leq \sqrt{\frac{2 \log(2/\delta)}{(1-\gamma)^2 \tau_E}}.$$

608 Applying a union bound, we further have that with probability at least  $1 - \delta$ , for all  $(Q, \pi) \in$   
609  $\mathcal{C}_\epsilon(\mathcal{Q} \times \Pi, \|\cdot\|_{\infty,1})$  it holds that

$$\left| \widehat{\mathcal{L}}(\pi, Q) - \mathcal{L}(\pi, Q) \right| \leq \sqrt{\frac{2 \log \left( 2 \mathcal{N}_\epsilon(\mathcal{Q} \times \Pi, \|\cdot\|_{\infty,1}) / \delta \right)}{(1-\gamma)^2 \tau_E}}.$$

610 Recall that  $\mathcal{C}_\epsilon(\mathcal{Q} \times \Pi, \|\cdot\|_{\infty,1})$  is assumed to be an  $\epsilon$ -covering set of the space  $\mathcal{Q} \times \Pi$  with  
611 respect to the norm  $(Q, \pi) \mapsto \|(Q, \pi)\|_{\infty,1} = \|Q\|_\infty + \max_{x \in \mathcal{X}} \|\pi(\cdot | x)\|_1$ . For any pair  
612  $(Q, \pi_k) \in \mathcal{Q} \times \Pi$ , let  $(Q_\epsilon, \pi_{k,\epsilon}) \in \mathcal{C}_\epsilon(\mathcal{Q} \times \Pi, \|\cdot\|_{\infty,1})$  denote the element of the covering such that  
613  $\|(Q, \pi_k) - (Q_\epsilon, \pi_{k,\epsilon})\|_{\infty,1} \leq \epsilon$ . Then, we have that

$$\begin{aligned} \left| \widehat{\mathcal{L}}(\pi_k, Q) - \widehat{\mathcal{L}}(\pi_{k,\epsilon}, Q_\epsilon) \right| &\leq \left| \frac{1}{\tau_E} \sum_{i=1}^{\tau_E} (Q(X_E^i, A_E^i) - Q_\epsilon(X_E^i, A_E^i)) \right| \\ &\quad + \left| \frac{1}{\tau_E} \sum_{i=1}^{\tau_E} \sum_{a \in \mathcal{A}} (\pi_{k,\epsilon}(a | X_E^i) Q_\epsilon(X_E^i, a) - \pi_k(a | X_E^i) Q(X_E^i, a)) \right| \\ &\leq \|Q - Q_\epsilon\|_\infty + \left| \frac{1}{\tau_E} \sum_{i=1}^{\tau_E} \sum_{a \in \mathcal{A}} (\pi_{k,\epsilon}(a | X_E^i) - \pi_k(a | X_E^i)) Q_\epsilon(X_E^i, a) \right| \\ &\quad + \left| \frac{1}{\tau_E} \sum_{i=1}^{\tau_E} \sum_{a \in \mathcal{A}} \pi_k(a | X_E^i) (Q(X_E^i, a) - Q_\epsilon(X_E^i, a)) \right|. \end{aligned}$$

614 Noting that for any  $Q \in \mathcal{Q}$ ,  $\|Q\|_\infty \leq \frac{1}{1-\gamma}$ , and that for any state  $x$ ,  $\pi_k(\cdot | x) \in \Delta(\mathcal{A})$ , using Hölder's  
615 inequality, we further have

$$\begin{aligned} \left| \widehat{\mathcal{L}}(\pi_k, Q) - \widehat{\mathcal{L}}(\pi_{k,\epsilon}, Q_\epsilon) \right| &\leq \|Q - Q_\epsilon\|_\infty + \frac{\max_{x \in \mathcal{X}} \|\pi_{k,\epsilon}(\cdot | x) - \pi_k(\cdot | x)\|_1}{1-\gamma} + \|Q - Q_\epsilon\|_\infty \\ &\leq \frac{2\epsilon}{1-\gamma}, \end{aligned}$$

where we used the definition of  $(\pi_{k,\epsilon}, Q_\epsilon)$  and  $\gamma \in (0, 1)$  in the last inequality. Similarly, for the true objective we have that

$$\begin{aligned}
|\mathcal{L}(\pi_k, Q) - \mathcal{L}(\pi_{k,\epsilon}, Q_\epsilon)| &\leq |\mathbb{E}_{(X,A) \sim \mu^{\pi_E}} [Q(X, A) - Q_\epsilon(X, A)]| \\
&\quad + |\mathbb{E}_{X \sim \nu^{\pi_E}} [Q(X, \pi_k) - Q_\epsilon(X, \pi_{k,\epsilon})]| \\
&\leq \|Q - Q_\epsilon\|_\infty + |\mathbb{E}_{X \sim \nu^{\pi_E}} [Q(X, \pi_k) - Q(X, \pi_{k,\epsilon})]| \\
&\quad + |\mathbb{E}_{X \sim \nu^{\pi_E}} [Q(X, \pi_{k,\epsilon}) - Q_\epsilon(X, \pi_{k,\epsilon})]| \\
&\leq \|Q - Q_\epsilon\|_\infty + \frac{\max_{x \in \mathcal{X}} \|\pi_{k,\epsilon}(\cdot | x) - \pi_k(\cdot | x)\|_1}{1 - \gamma} + \|Q - Q_\epsilon\|_\infty \\
&\leq \frac{2\epsilon}{1 - \gamma}.
\end{aligned}$$

Therefore, with probability at least  $1 - \delta$ , it holds that for any  $k \in [K]$  and any  $Q \in \mathcal{Q}$ ,

$$\begin{aligned}
|\widehat{\mathcal{L}}(\pi_k, Q) - \mathcal{L}(\pi_k, Q)| &\leq |\widehat{\mathcal{L}}(\pi_k, Q) - \widehat{\mathcal{L}}(\pi_{k,\epsilon}, Q_\epsilon)| + |\widehat{\mathcal{L}}(\pi_{k,\epsilon}, Q_\epsilon) - \mathcal{L}(\pi_{k,\epsilon}, Q_\epsilon)| \\
&\quad + |\mathcal{L}(\pi_k, Q) - \mathcal{L}(\pi_{k,\epsilon}, Q_\epsilon)| \\
&\leq \frac{4\epsilon}{1 - \gamma} + \sqrt{\frac{2 \log(2\mathcal{N}_\epsilon(\mathcal{Q} \times \Pi, \|\cdot\|_{\infty,1})/\delta)}{(1 - \gamma)^2 \tau_E}}.
\end{aligned}$$

Moreover, since the above bound holds for all  $Q \in \mathcal{Q}$ , it holds for the supremum over this class. With probability at least  $1 - \delta$ , we have for any  $k \in [K]$  that

$$\sup_{Q \in \mathcal{Q}} |\widehat{\mathcal{L}}(\pi_k, Q) - \mathcal{L}(\pi_k, Q)| \leq \frac{4\epsilon}{1 - \gamma} + \sqrt{\frac{2 \log(2\mathcal{N}_\epsilon(\mathcal{Q} \times \Pi, \|\cdot\|_{\infty,1})/\delta)}{(1 - \gamma)^2 \tau_E}}.$$

The proof is concluded by noting that the above proof holds for any covering size  $\epsilon > 0$ .  $\square$

#### B.4 Proof of Lemma 5 (concentration linear case)

We now instantiate Lemma 9 in the linear  $Q^\pi$ -realizable setting. For this purpose, we compute a bound on the covering number of the class  $\Pi_{\text{lin}}$ , defined in Equation (3).

**Lemma 10** (Covering number of  $\Pi_{\text{lin}}$ ). *It holds that the covering number of the policy class  $\Pi_{\text{lin}}$  can be bounded as*

$$\mathcal{N}_\epsilon(\Pi_{\text{lin}}, \|\cdot\|_1) \leq \left(1 + \frac{2K\eta B_\theta B_\varphi A}{\epsilon}\right)^d,$$

where, with a slight abuse of notation,  $\|\cdot\|_1$  denotes the norm defined for any  $\pi \in \Pi_{\text{lin}}$  as  $\|\pi\|_1 = \sup_{x \in \mathcal{X}} \|\pi(\cdot | x)\|_1$ . Moreover, let

$$\mathcal{Q}_{\text{lin}} = \{Q: \mathcal{X} \times \mathcal{A} \rightarrow \mathbb{R} : \exists \theta \in \mathfrak{B}(B_\theta), \forall (x, a) \in \mathcal{X} \times \mathcal{A}, Q(x, a) = \langle \theta, \varphi(x, a) \rangle\}$$

be the class of linear action-value functions. Then, it holds that

$$\mathcal{N}_\epsilon(\mathcal{Q}_{\text{lin}} \times \Pi_{\text{lin}}, \|\cdot\|_{\infty,1}) \leq \left(1 + \frac{4K\eta B_\theta B_\varphi A}{\epsilon}\right)^{2d}.$$

*Proof.* Let us consider two policies  $\pi$  and  $\pi'$  in the class  $\Pi_{\text{lin}}$ . There exist  $\theta_1, \dots, \theta_K \in \mathfrak{B}(B_\theta)$  and  $\theta'_1, \dots, \theta'_K \in \mathfrak{B}(B_\theta)$  such that for any state-action pair  $(x, a) \in \mathcal{X} \times \mathcal{A}$ ,  $\pi$  and  $\pi'$  can be written as

$$\pi(a | x) = \frac{\exp\left(\eta \left\langle \varphi(x, a), \sum_{k=1}^K \theta_k \right\rangle\right)}{\sum_{b \in \mathcal{A}} \exp\left(\eta \left\langle \varphi(x, b), \sum_{k=1}^K \theta_k \right\rangle\right)},$$

and

$$\pi'(a | x) = \frac{\exp\left(\eta \left\langle \varphi(x, a), \sum_{k=1}^K \theta'_k \right\rangle\right)}{\sum_{b \in \mathcal{A}} \exp\left(\eta \left\langle \varphi(x, b), \sum_{k=1}^K \theta'_k \right\rangle\right)}.$$



633 In particular, let us fix a state  $x \in \mathcal{X}$ , and denote  $\bar{\theta}_K = \sum_{k=1}^K \theta_k$ ,  $\bar{\theta}'_K = \sum_{k=1}^K \theta'_k$ . First, by  
 634 Cauchy-Schwartz's inequality, we have

$$\|\pi(\cdot | x) - \pi'(\cdot | x)\|_1 \leq \sqrt{A} \|\pi(\cdot | x) - \pi'(\cdot | x)\|.$$

635 By 1-Lipshitzness of the softmax function (Lemma 16), it holds that

$$\begin{aligned} \|\pi(\cdot | x) - \pi'(\cdot | x)\|_1 &\leq \eta \sqrt{A} \|\langle \varphi(x, \cdot), \bar{\theta}_K - \bar{\theta}'_K \rangle\| \\ &= \eta \sqrt{A \sum_{a \in \mathcal{A}} (\langle \varphi(x, a), \bar{\theta}_K - \bar{\theta}'_K \rangle)^2} \\ &\leq \eta \sqrt{A \sum_{a \in \mathcal{A}} \|\varphi(x, a)\|^2 \|\bar{\theta}_K - \bar{\theta}'_K\|^2} \quad (\text{Cauchy-Schwartz}) \\ &\leq \eta B_\varphi A \|\bar{\theta}_K - \bar{\theta}'_K\| \quad (\text{Assumption 1}) \\ &\leq \eta B_\varphi A \sum_{k=1}^K \|\theta_k - \theta'_k\| \quad (\text{Triangle inequality}) \\ &\leq K \eta B_\varphi A \max_{k \in [K]} \|\theta_k - \theta'_k\|. \end{aligned}$$

636 Therefore, the  $\epsilon$ -covering number for  $\Pi_{\text{lin}}$  with respect to the norm  $\|\cdot\|_1$ ,  $\mathcal{N}_\epsilon(\Pi_{\text{lin}}, \|\cdot\|_1)$ , is upper-  
 637 bounded by the  $\frac{\epsilon}{K \eta B_\varphi A}$ -covering number of the Euclidean ball  $\mathfrak{B}(B_\theta)$  with respect to the norm  $\|\cdot\|$ ,  
 638 and

$$\begin{aligned} \mathcal{N}_\epsilon(\Pi_{\text{lin}}, \|\cdot\|_1) &\leq \mathcal{N}_{\frac{\epsilon}{K \eta B_\varphi A}}(\mathfrak{B}(B_\theta), \|\cdot\|) \\ &\leq \left(1 + \frac{2K \eta B_\theta B_\varphi A}{\epsilon}\right)^d, \end{aligned}$$

639 where we used Lemma 17 in the last inequality. For the second part of the lemma, let us consider  
 640  $Q, Q' \in \mathcal{Q}_{\text{lin}}$ . By definition of  $\mathcal{Q}_{\text{lin}}$ , there exists  $\theta, \theta' \in \mathfrak{B}(B_\theta)$  such that for any state-action pair  
 641  $(x, a)$ ,  $Q(x, a) = \langle \varphi(x, a), \theta \rangle$  and  $Q'(x, a) = \langle \varphi(x, a), \theta' \rangle$ . Then,

$$\max_{x, a \in \mathcal{X} \times \mathcal{A}} |Q(x, a) - Q'(x, a)| = \max_{x, a \in \mathcal{X} \times \mathcal{A}} |\langle \varphi(x, a), \theta - \theta' \rangle| \leq B_\varphi \|\theta - \theta'\|.$$

642 Therefore, the  $\epsilon$ -covering number of  $\mathcal{Q}_{\text{lin}}$ ,  $\mathcal{N}_\epsilon(\mathcal{Q}_{\text{lin}}, \|\cdot\|_\infty)$ , is upper-bounded by the  $\epsilon/B_\varphi$ -covering  
 643 number of the  $d$ -dimensional ball with radius  $B_\theta$ ,  $\mathcal{N}_{\epsilon/B_\varphi}(\mathfrak{B}(B_\theta), \|\cdot\|)$ . We have

$$\mathcal{N}_\epsilon(\mathcal{Q}_{\text{lin}}, \|\cdot\|_\infty) \leq \mathcal{N}_{\epsilon/B_\varphi}(\mathfrak{B}(B_\theta), \|\cdot\|) \leq \left(1 + \frac{2B_\theta B_\varphi}{\epsilon}\right)^d.$$

644 Finally, the proof is concluded by noting that

$$\mathcal{N}_\epsilon(\mathcal{Q}_{\text{lin}} \times \Pi_{\text{lin}}, \|\cdot\|_{\infty, 1}) \leq \mathcal{N}_{\epsilon/2}(\Pi_{\text{lin}}, \|\cdot\|_1) \mathcal{N}_{\epsilon/2}(\mathcal{Q}_{\text{lin}}, \|\cdot\|_\infty).$$

645 □

646 Finally, the following result proves the concentration of the estimators used in Algorithm 1.

647 **Lemma 11.** *Let  $\{\pi_k\}_{k \in [K]}$  be the sequence of policies generated by Algorithm 1 and let  $\Delta(\pi_k)$  be*  
 648 *defined as in Proposition 1. Then, with probability at least  $1 - \delta$ , it holds that*

$$\sum_{k=1}^K \Delta(\pi_k) \leq 1 + 2K \sqrt{\frac{8d}{(1-\gamma)^2 \tau_E} \log \left( \frac{2 + 16B_\theta B_\varphi K}{(1-\gamma)\delta} \right)}.$$

649 *Proof.* By Lemma 9, it holds that

$$\begin{aligned}
\sum_{k=1}^K \Delta(\pi_k) &\leq K \inf_{\epsilon: \epsilon > 0} \left\{ \frac{4\epsilon}{1-\gamma} + 2\sqrt{\frac{2\log\left(2\mathcal{N}_\epsilon\left(\mathcal{Q}_{\text{lin}} \times \Pi_{\text{lin}}, \|\cdot\|_{\infty,1}\right)/\delta\right)}{(1-\gamma)^2\tau_E}} \right\} \\
&\leq 1 + 2K\sqrt{\frac{2\log\left(2\mathcal{N}_{(1-\gamma)/4K}\left(\mathcal{Q}_{\text{lin}} \times \Pi_{\text{lin}}, \|\cdot\|_{\infty,1}\right)/\delta\right)}{(1-\gamma)^2\tau_E}} \\
&\leq 1 + 2K\sqrt{\frac{2}{(1-\gamma)^2\tau_E} \log\left(\frac{2}{\delta}\left(1 + \frac{16K^2\eta B_\theta B_\varphi A}{1-\gamma}\right)^{2d}\right)} \\
&\leq 1 + 4K\sqrt{\frac{d}{(1-\gamma)^2\tau_E} \log\left(\frac{2 + 32K^2\eta B_\theta B_\varphi A}{(1-\gamma)\delta}\right)},
\end{aligned}$$

650 where the third inequality follows from Lemma 10.  $\square$

## 651 B.5 Proof of Theorem 2 (sample complexity guarantee for linear $Q^\pi$ -realizable MDPs)

652 **Theorem 2.** Let Assumption 1 hold. Run Algorithm 1 for  $K = \frac{2\log|\mathcal{A}|}{(1-\gamma)^2\epsilon^2}$  iterations, with a learning  
653 rate  $\eta = (1-\gamma)\sqrt{2\log|\mathcal{A}|/K}$ , and  $\tau_E = \mathcal{O}\left(\frac{d}{(1-\gamma)^2\epsilon^2} \log\left(\frac{B_\theta B_\varphi \log|\mathcal{A}|}{(1-\gamma)\epsilon}\right)\right)$  samples collected by  
654 any expert policy  $\pi_E$ . Then, the output satisfies  $\mathbb{E}[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] \leq 5\epsilon$ .

655 *Proof.* By Proposition 1, we have

$$\mathbb{E}[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] \leq \frac{1}{K} \sum_{k=1}^K \mathbb{E}[\mathcal{L}(\pi_k; Q_k)] + \frac{2}{K} \sum_{k=1}^K \mathbb{E}[\Delta(\pi_k)].$$

656 Using Lemma 4 with a learning rate of  $\eta = (1-\gamma)\sqrt{\frac{2\log|\mathcal{A}|}{K}}$  and dividing by  $K$ , we obtain that

$$\frac{1}{K} \sum_{k=1}^K \mathcal{L}(\pi_k; Q_k) \leq \sqrt{\frac{2\log|\mathcal{A}|}{(1-\gamma)^2 K}}.$$

657 Therefore, setting  $K = \frac{2\log|\mathcal{A}|}{(1-\gamma)^2\epsilon^2}$  guarantees  $\frac{1}{K} \sum_{k=1}^K \mathcal{L}(\pi_k; Q_k) \leq \epsilon$ . Then, using the high-  
658 probability bound in Lemma 5 and the fact that  $\frac{1}{K} \sum_{k=1}^K \Delta(\pi_k)$  is a random variable  
659 bounded by  $2(1-\gamma)^{-1}$  almost surely, we obtain the following expectation bound which holds for all  
660  $\delta > 0$ ,

$$\frac{1}{K} \sum_{k=1}^K \mathbb{E}[\Delta(\pi_k)] \leq \frac{1}{K} + C\sqrt{\frac{d}{(1-\gamma)^2\tau_E} \log\left(\frac{B_\theta B_\varphi A}{(1-\gamma)\delta\epsilon}\right)} + \frac{2\delta}{1-\gamma},$$

661 for some  $C \in \mathbb{R}$ . Note that the choice of parameters ensures  $\frac{1}{K} \leq \frac{\epsilon}{2}$ . Setting  $\delta = \frac{\epsilon(1-\gamma)}{4}$  and

$$\tau_E \geq \frac{C^2 d}{(1-\gamma)^2\epsilon^2} \log\left(\frac{B_\theta B_\varphi A}{(1-\gamma)\delta\epsilon}\right)$$

662 this bound implies that  $\frac{2}{K} \sum_{k=1}^K \mathbb{E}[\Delta(\pi_k)] \leq 4\epsilon$ . Thus, we conclude that  $\mathbb{E}[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] \leq 5\epsilon$ .  $\square$

## 663 B.6 Proof of Lemma 6 (concentration nonlinear case)

664 Before presenting the proof of Theorem 3, we provide a bound on the covering number of the class  
665  $\mathcal{Q} \times \Pi_{\mathcal{Q}}$ , where  $\Pi_{\mathcal{Q}}$  is defined in Equation (4). It turns out that the covering number of this class  
666 is exponential in  $K$ . In the linear case, the exponential dependence in  $K$  was avoided because the  
667 state-action value class is closed under addition.

**Lemma 12** (Covering number of  $\Pi_{\mathcal{Q}}$ ). *It holds that the covering number of the policy class  $\Pi_{\mathcal{Q}}$  can be bounded as*

$$\mathcal{N}_{\epsilon}(\Pi_{\mathcal{Q}}, \|\cdot\|_1) \leq \mathcal{N}_{\frac{\epsilon}{K\eta A}}(\mathcal{Q}, \|\cdot\|_{\infty})^K,$$

where, with a slight abuse of notation,  $\|\cdot\|_1$  denotes the norm defined for any  $\pi \in \Pi_{\mathcal{Q}}$  as  $\|\pi\|_1 = \sup_{x \in \mathcal{X}} \|\pi(\cdot | x)\|_1$ . Moreover,

$$\mathcal{N}_{\epsilon}(\mathcal{Q} \times \Pi_{\mathcal{Q}}, \|\cdot\|_{\infty,1}) \leq \mathcal{N}_{\frac{\epsilon}{K\eta A}}(\mathcal{Q}, \|\cdot\|_{\infty})^{K+1}.$$

*Proof.* Let us consider two policies  $\pi$  and  $\pi'$  in the class  $\Pi_{\mathcal{Q}}$ . There exist  $Q_1, \dots, Q_K \in \mathcal{Q}$  and  $Q'_1, \dots, Q'_K \in \mathcal{Q}$  such that for any state-action pair  $(x, a) \in \mathcal{X} \times \mathcal{A}$ ,  $\pi$  and  $\pi'$  can be written as

$$\pi(a | x) = \frac{\exp\left(\eta \sum_{k=1}^K Q_k(x, a)\right)}{\sum_{b \in \mathcal{A}} \exp\left(\eta \sum_{k=1}^K Q_k(x, b)\right)},$$

and

$$\pi'(a | x) = \frac{\exp\left(\eta \sum_{k=1}^K Q'_k(x, a)\right)}{\sum_{b \in \mathcal{A}} \exp\left(\eta \sum_{k=1}^K Q'_k(x, b)\right)}.$$

Let  $x \in \mathcal{X}$ . Using  $\|\cdot\|_1 \leq \sqrt{A} \|\cdot\|$  in  $\mathbb{R}^A$  and by 1-Lipshitzness of the softmax function (Lemma 16), it holds that

$$\begin{aligned} \|\pi(\cdot | x) - \pi'(\cdot | x)\|_1 &\leq \sqrt{A} \|\pi(\cdot | x) - \pi'(\cdot | x)\| \\ &\leq \eta \sqrt{A} \left\| \sum_{k=1}^K (Q_k(x, \cdot) - Q'_k(x, \cdot)) \right\| \\ &\leq \eta \sqrt{A} \sum_{k=1}^K \|Q_k(x, \cdot) - Q'_k(x, \cdot)\| && \text{(Triangle inequality)} \\ &\leq \eta A \sum_{k=1}^K \sup_{a \in \mathcal{A}} |Q_k(x, a) - Q'_k(x, a)| && (\|\cdot\| \leq \sqrt{A} \|\cdot\|_{\infty}) \\ &\leq \eta A \sup_{x \in \mathcal{X}} \left\{ \sum_{k=1}^K \sup_{a \in \mathcal{A}} |Q_k(x, a) - Q'_k(x, a)| \right\} \\ &\leq \eta A \sum_{k=1}^K \|Q_k - Q'_k\|_{\infty} && \text{(Triangle inequality)}. \end{aligned}$$

In particular, this implies

$$\max_{x \in \mathcal{X}} \|\pi(\cdot | x) - \pi'(\cdot | x)\|_1 \leq \eta A \sum_{k=1}^K \|Q'_k - Q_k\|_{\infty}.$$

Thus, the  $\epsilon$ -covering number for  $\Pi_{\mathcal{Q}}$ ,  $\mathcal{N}_{\epsilon}(\Pi_{\mathcal{Q}}, \|\cdot\|_1)$ , is upper-bounded by the  $\frac{\epsilon}{K\eta A}$ -covering number of the class  $\mathcal{Q}$  to the power  $K$ , i.e.,  $\mathcal{N}_{\frac{\epsilon}{K\eta A}}(\mathcal{Q}, \|\cdot\|_{\infty})^K$ . Thus,

$$\mathcal{N}_{\epsilon}(\Pi_{\mathcal{Q}}, \|\cdot\|_1) \leq \mathcal{N}_{\frac{\epsilon}{K\eta A}}(\mathcal{Q}, \|\cdot\|_{\infty})^K.$$

The proof is concluded by noting that the covering number increases with the precision (when  $\epsilon$  decreases), and therefore, we can write

$$\begin{aligned} \mathcal{N}_{\epsilon}(\mathcal{Q} \times \Pi_{\mathcal{Q}}, \|\cdot\|_{\infty,1}) &\leq \mathcal{N}_{\epsilon/2}(\mathcal{Q}, \|\cdot\|_{\infty}) \mathcal{N}_{\epsilon/2}(\Pi_{\mathcal{Q}}, \|\cdot\|_1) \\ &\leq \mathcal{N}_{\epsilon/2}(\mathcal{Q}, \|\cdot\|_{\infty}) \mathcal{N}_{\frac{\epsilon}{K\eta A}}(\mathcal{Q}, \|\cdot\|_{\infty})^K \\ &\leq \mathcal{N}_{\frac{\epsilon}{K\eta A}}(\mathcal{Q}, \|\cdot\|_{\infty})^{K+1}. \end{aligned}$$

□

Finally, the following result proves the concentration of the estimators used in Algorithm 2.

**Lemma 13.** *Let  $\{\pi_k\}_{k \in [K]}$  be the sequence of policies generated by Algorithm 2. Then, with probability at least  $1 - \delta$ , for any  $k \in [K]$ , it holds that*

$$\sup_{Q \in \mathcal{Q}} \left| \widehat{\mathcal{L}}(\pi_k, Q) - \mathcal{L}(\pi_k, Q) \right| \leq \frac{1}{2K} + \sqrt{\frac{2(K+1) \log(2\mathcal{N}_{(1-\gamma)/8K}(\mathcal{Q}, \|\cdot\|_\infty)/\delta)}{(1-\gamma)^2 \tau_E}}.$$

*Proof.* Note that by construction, the policy sequence  $\{\pi_k\}_{k \in [K]}$  generated by Algorithm 2 belongs to the policy class  $\Pi_{\mathcal{Q}}$ . Therefore, invoking Lemma 9, we have that with probability at least  $1 - \delta$ , for any  $k \in [K]$ , it holds that

$$\Delta(\pi_k) \leq \inf_{\epsilon: \epsilon > 0} \left\{ \frac{4\epsilon}{1-\gamma} + \sqrt{\frac{2 \log(2\mathcal{N}_\epsilon(\mathcal{Q} \times \Pi_{\mathcal{Q}}, \|\cdot\|_{\infty,1})/\delta)}{(1-\gamma)^2 \tau_E}} \right\}.$$

Therefore, choosing  $\epsilon = \frac{1-\gamma}{8K}$ , we get

$$\begin{aligned} \Delta(\pi_k) &\leq \frac{1}{2K} + \sqrt{\frac{2 \log(2\mathcal{N}_{(1-\gamma)/8K}(\mathcal{Q} \times \Pi_{\mathcal{Q}}, \|\cdot\|_{\infty,1})/\delta)}{(1-\gamma)^2 \tau_E}} \\ &\leq \frac{1}{2K} + \sqrt{\frac{2(K+1) \log(2\mathcal{N}_{\frac{1-\gamma}{8K^2\eta A}}(\mathcal{Q}, \|\cdot\|_\infty)/\delta)}{(1-\gamma)^2 \tau_E}}. \end{aligned}$$

□

## B.7 Proof of Theorem 3 (sample complexity guarantee for $Q^\pi$ -realizable MDPs)

We are now ready for the proof of Theorem 3, which we restate for convenience.

**Theorem 3.** *Let Assumption 2 hold. Run Algorithm 2 for  $K = \frac{2 \log |\mathcal{A}|}{(1-\gamma)^2 \varepsilon^2}$  iterations, with a learning rate  $\eta = (1-\gamma) \sqrt{2 \log |\mathcal{A}| / K}$  and  $\tau_E = \mathcal{O}\left(\frac{\log |\mathcal{A}|}{(1-\gamma)^2 \varepsilon^4} \log\left(\frac{\mathcal{N}_{\epsilon'}(\mathcal{Q}, \|\cdot\|_\infty)}{\varepsilon(1-\gamma)}\right)\right)$  samples collected by any expert policy  $\pi_E$ , where  $\epsilon' = \frac{(1-\gamma)^3 \varepsilon^2}{4 \log |\mathcal{A}|}$ . Then, the output satisfies  $\mathbb{E}[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] = \mathcal{O}(\varepsilon)$ .*

*Proof.* Recall that by Proposition 1, we have

$$\mathbb{E}[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}] \leq \frac{1}{K} \sum_{k=1}^K \mathbb{E}[\mathcal{L}(\pi_k; Q_k)] + \frac{2}{K} \sum_{k=1}^K \mathbb{E}[\Delta(\pi_k)].$$

Then, by Lemma 4, it holds that

$$\frac{1}{K} \sum_{k=1}^K \mathbb{E}[\mathcal{L}(\pi_k; Q_k)] \leq \frac{\log(A)}{\eta K} + \frac{\eta}{(1-\gamma)^2}.$$

Moreover, by Lemma 13, with probability at least  $1 - \delta$ , it holds that

$$2 \sum_{k=1}^K \Delta(\pi_k) \leq 1 + 2K \sqrt{\frac{2(K+1) \log(2\mathcal{N}_{\frac{1-\gamma}{8K^2\eta A}}(\mathcal{Q}, \|\cdot\|_\infty)/\delta)}{(1-\gamma)^2 \tau_E}}.$$

Since  $\frac{1}{K} \sum_{k=1}^K \Delta(\pi_k)$  is bounded almost surely by  $4(1-\gamma)^{-1}$ , we have that for any  $\delta > 0$

$$\frac{2}{K} \sum_{k=1}^K \mathbb{E}[\Delta(\pi_k)] \leq \frac{1}{K} + 2 \sqrt{\frac{2(K+1) \log(2\mathcal{N}_{\frac{1-\gamma}{8K^2\eta A}}(\mathcal{Q}, \|\cdot\|_\infty)/\delta)}{(1-\gamma)^2 \tau_E}} + \frac{4\delta}{1-\gamma}.$$

700 Setting  $\eta = (1 - \gamma)\sqrt{2\log(A)/K}$ , we get

$$\mathbb{E}\left[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}\right] \leq \sqrt{\frac{2\log(A)}{(1-\gamma)^2 K}} + \frac{1}{K} + 2\sqrt{\frac{2(K+1)\log(2\mathcal{N}_{\epsilon'}(\mathcal{Q}, \|\cdot\|_\infty)/\delta)}{(1-\gamma)^2 \tau_E}} + \frac{4\delta}{1-\gamma},$$

701 where we denoted  $\epsilon' = \frac{1}{\sqrt{28K^{3/2}\sqrt{\log(A)A}}}$ . Setting  $\delta = \frac{(1-\gamma)\epsilon}{4}$  and  $K = \frac{2\log A}{(1-\gamma)^2 \epsilon^2}$ , we further have

$$\mathbb{E}\left[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}\right] \leq \epsilon + \frac{\epsilon}{2} + 4\sqrt{\frac{\log(A)}{(1-\gamma)^4 \epsilon^2 \tau_E} \log\left(\frac{2\mathcal{N}_{\epsilon'}(\mathcal{Q}, \|\cdot\|_\infty)}{\delta}\right)}.$$

702 Finally, setting

$$\tau_E \geq \frac{16\log(A)}{(1-\gamma)^4 \epsilon^4} \log\left(\frac{2\mathcal{N}_{\epsilon'}(\mathcal{Q}, \|\cdot\|_\infty)}{\delta}\right),$$

703 we guarantee that

$$\mathbb{E}\left[\rho^{\pi_E} - \rho^{\pi^{\text{out}}}\right] = \mathcal{O}(\epsilon).$$

704

□

## 705 C Technical tools

706 **Lemma 14** (Hoeffding's inequality). *Let  $X_1, \dots, X_n$  be independent random variables such that*  
 707  *$|X_i| \leq M$  for all  $i$ . Then, for any  $\epsilon > 0$ ,*

$$\mathbb{P}\left(\left|\frac{1}{n} \sum_{i=1}^n (X_i - \mathbb{E}(X_i))\right| > \epsilon\right) \leq 2e^{-\frac{2n\epsilon^2}{M^2}}.$$

708 **Lemma 15** (Simplified version of [Orabona, 2023](#), Theorem 6.10). *Let us consider a non-empty*  
 709 *closed convex set  $V$ , an arbitrary sequence of adaptively chosen loss vectors  $(\ell_k)_{k=1}^K$  such that*  
 710  *$\|\ell_k\|_\infty \leq \ell_{\max}$ , and let  $D : V \times \text{int}(V) \rightarrow \mathbb{R}$  be a Bregman divergence induced by a  $\lambda$ -strongly*  
 711 *convex function in the  $\ell_1$ -norm. Then, for all  $u \in V$ , the sequence  $(x_k)_{k=1}^K$  generated for any  $k$  as*

$$x_{k+1} = \arg \min_{v \in V} \left\{ \langle \ell_k, v \rangle + \frac{1}{\eta} D(v, x_k) \right\}$$

712 *for an arbitrary initial  $x_1$  satisfies*

$$\sum_{k=1}^K \langle \ell_k, x_k - u \rangle \leq \frac{D(u, x_1)}{\eta} + \frac{\eta K \ell_{\max}^2}{2\lambda}.$$

713 **Lemma 16** ([Gao and Pavel, 2017](#), Proposition 4). *For any  $\eta > 0$ , let the softmax function be defined*  
 714 *for any  $z \in \mathbb{R}^n$  as*

$$\text{softmax}(z) = \left( \frac{e^{\eta z_i}}{\sum_{j=1}^n e^{\eta z_j}} \right)_{i \in [n]}.$$

715 *Then, the softmax function is  $\eta$ -Lipschitz with respect to  $\|\cdot\|_2$ . That is, for any  $z, z' \in \mathbb{R}^n$ , we have*

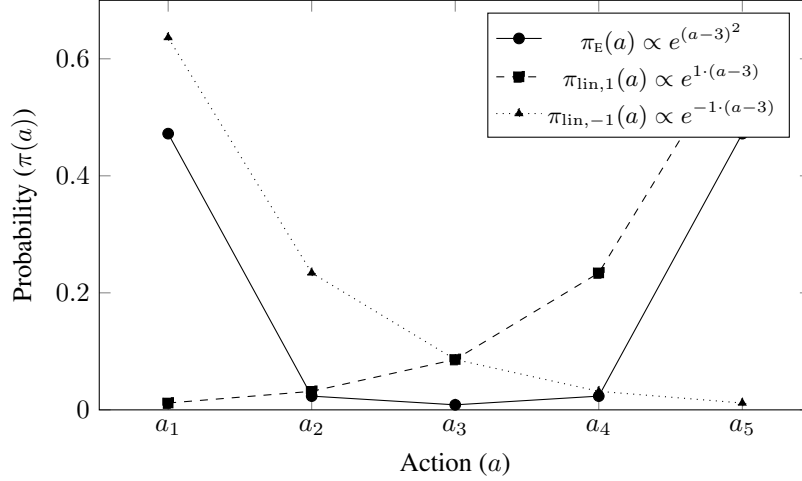
$$\|\text{softmax}(z) - \text{softmax}(z')\|_2 \leq \eta \|z - z'\|_2.$$

716 **Lemma 17** (Covering number of a Euclidean ball). *The covering number of the Euclidean ball of*  
 717 *radius  $R$  in  $\mathbb{R}^d$ ,  $\mathfrak{B}(R)$ , is*

$$\mathcal{N}_\epsilon(\mathfrak{B}(R), \|\cdot\|) \leq \left(1 + \frac{2R}{\epsilon}\right)^d.$$



Figure 3: Comparison of linear and quadratic softmax policies with  $A = 5$  actions and features  $\varphi(a) = a - 3$ .



## D On the guarantees of misspecified BC in linear $Q^\pi$ -realizable MDPs

It is natural to question whether existing bounds for behavioral cloning (BC) in misspecified settings [e.g., Rohatgi et al., 2025, Foster et al., 2024] offer satisfactory sample complexity guarantees for imitating an arbitrarily complex expert within a linear  $Q^\pi$ -realizable MDP. This section presents a negative result, demonstrating that the approximation error incurred by BC, when restricted to a linear softmax policy class (denoted  $\Pi_{\text{lin}}$ ), can be large even in a simple linear  $Q^\pi$ -realizable MDP.

Consider a single-state MDP defined as follows. Let  $A \in \mathbb{N}^*$  be the number of actions, with the action space  $\mathcal{A} = \llbracket 1, \dots, A \rrbracket$ . For each action  $a \in \mathcal{A}$ , there is a scalar feature  $\varphi(a) = -\frac{|A|}{2} + a \in \mathbb{R}$ . To ensure the MDP is linear  $Q^\pi$ -realizable, the true reward function is  $r_{\text{true}}(a) = \zeta \varphi(a)$  for some parameter  $\zeta \in \mathbb{R}$  unknown to the learner. We define a *softmax quadratic* expert policy  $\pi_E$  as

$$\pi_E(a) = \frac{\exp(\varphi(a)^2)}{\sum_{b \in \mathcal{A}} \exp(\varphi(b)^2)}.$$

This expert policy assigns the highest probability to extremal actions (i.e.,  $a = 1$  and  $a = A$ ). In contrast, linear softmax policies  $\pi \in \Pi_{\text{lin}}$  (which are commonly used for BC in feature-based settings) are inherently designed to produce monotonic probability distributions over the action space when features are ordered (i.e., for actions  $a, a' \in \mathcal{A}$  with  $a' > a$ , either  $\pi(a) \leq \pi(a')$  or  $\pi(a) \geq \pi(a')$ ). Consequently, for  $A > 2$ , no policy in  $\Pi_{\text{lin}}$  can achieve a small Hellinger distance to this softmax quadratic expert. We illustrate this in Figure 3, where we compare the softmax quadratic expert with two linear softmax policies. Due to the monotonicity constraint, the linear softmax policies are unable to approximate the expert policy everywhere.

It remains an open question whether behavioral cloning analyses can be refined to better leverage the underlying MDP structure in such misspecified scenarios. Specifically, for the constructed example, it would be advantageous if the misspecification error in existing bounds were characterized in terms of feature expectations (e.g.,  $\sum_{a \in \mathcal{A}} \pi(a) \varphi(a)$ ) rather than state-action distributions.

## E Experimental details

For the first experiment shown in Figure 1, one may wonder if the underperformance of behavioural cloning might be due to underoptimizing the empirical log-likelihood. We have ruled out this possibility by going into great lengths to optimize the likelihood, and in fact the log-likelihood has approached its minimum value of zero very closely in our experiment (meaning that the probability assigned to the actions seen in the expert dataset is almost 1). For this optimization task, we have used Adam with default parameter settings. For the experiments in Figure 2, algorithms are implemented using a shared

neural network architecture consisting of 3 layers with 64 neurons per layer. This architecture matches the one used for experiments in the same environments by Garg et al. [2021]. For behavioral cloning, we employ a separate three-layer multilayer perceptron with 128 neurons per layer. Implementations of IQ-Learn and P<sup>2</sup>IL utilize their original hyperparameter configurations as reported in their respective publications. All networks are optimized using the Adam optimizer [Kingma and Ba, 2014] with a learning rate of  $5 \times 10^{-3}$  and default momentum parameters ( $\beta_1 = 0.9, \beta_2 = 0.999$ ). The implementations are built using PyTorch [Paszke, 2019].

For algorithms with a primal-dual structure (*i.e.*, IQ-Learn, P<sup>2</sup>IL, and SP0IL), the policy update is performed using a Soft DQN-style update [cf. Haarnoja et al., 2017] with a fixed temperature parameter. These three algorithms thus only differ in terms of their Q-value updates, and thus this experiment serves to assess the effectiveness of the novel critic loss introduced in this work.

## F Future directions

**Possible improvements.** The most interesting immediate question that one can ask about our result is if the  $O(\varepsilon^{-4})$  scaling featured in our general bound is improvable under the conditions we assume. We believe that substantially different algorithmic and analytic ideas would be necessary to answer this question, but we also think that our primal-dual framework provides a good starting point towards making such improvements. Furthermore, we would be curious to investigate appropriate notions of misspecification that our algorithm can deal with. It can be easily shown that requiring  $Q^\pi$ -realizability only up to a worst-case additive error of order  $\varepsilon_{\text{approx}}$  would incur the same additional term in the error bounds, but we believe that this assumption is too strong to warrant interest and we did not include an explicit statement. A much more interesting question is if this approximation guarantee would only be required to hold locally in the state-action pairs visited by the expert, or only for specific policies (most ideally only the expert policy). Given the numerous negative results in RL theory about such weaker function approximators, we are not optimistic that these latter improvements are possible, but nevertheless (and once again), we feel that our analytic framework can provide suitable tools for analyzing such questions.

**Learning from features only.** In the case of linear function approximation, the current approach critically relies on observing the expert state-action pairs to compute the vectors  $\{\hat{g}_k\}_{k=1}^K$ . It would be interesting to check if an alternative algorithm can achieve the same guarantees by only observing the feature vectors instead. In other words, the design of an algorithm taking as input a dataset  $\{\varphi(X_E^i, A_E^i)\}_{i=1}^{\tau_E}$  is an interesting open problem.

**New efficient algorithms that learn state-action value functions from expert data.** Despite having proven successful in practice [Garg et al., 2021], the idea of learning a state-action value function from expert data without passing through a learned reward function has not been used to develop theoretically grounded algorithms. Our work is the first example of an algorithm enjoying theoretical guarantees applying this principle. We expect this principle to find other applications in imitation learning theory, for example on the open problem of learning to imitate an expert from state-only trajectory given trajectory access to a linear- $Q^\pi$  realizable MDP.

## NeurIPS Paper Checklist

The checklist is designed to encourage best practices for responsible machine learning research, addressing issues of reproducibility, transparency, research ethics, and societal impact. Do not remove the checklist: **The papers not including the checklist will be desk rejected.** The checklist should follow the references and follow the (optional) supplemental material. The checklist does NOT count towards the page limit.

Please read the checklist guidelines carefully for information on how to answer these questions. For each question in the checklist:

- You should answer [Yes], [No], or [NA].
- [NA] means either that the question is Not Applicable for that particular paper or the relevant information is Not Available.
- Please provide a short (1–2 sentence) justification right after your answer (even for NA).

**The checklist answers are an integral part of your paper submission.** They are visible to the reviewers, area chairs, senior area chairs, and ethics reviewers. You will be asked to also include it (after eventual revisions) with the final version of your paper, and its final version will be published with the paper.

The reviewers of your paper will be asked to use the checklist as one of the factors in their evaluation. While "[Yes]" is generally preferable to "[No]", it is perfectly acceptable to answer "[No]" provided a proper justification is given (e.g., "error bars are not reported because it would be too computationally expensive" or "we were unable to find the license for the dataset we used"). In general, answering "[No]" or "[NA]" is not grounds for rejection. While the questions are phrased in a binary way, we acknowledge that the true answer is often more nuanced, so please just use your best judgment and write a justification to elaborate. All supporting evidence can appear either in the main paper or the supplemental material, provided in appendix. If you answer [Yes] to a question, in the justification please point to the section(s) where related material for the question can be found.

IMPORTANT, please:

- **Delete this instruction block, but keep the section heading “NeurIPS Paper Checklist”,**
- **Keep the checklist subsection headings, questions/answers and guidelines below.**
- **Do not modify the questions and only use the provided macros for your answers.**

### 1. Claims

Question: Do the main claims made in the abstract and introduction accurately reflect the paper’s contributions and scope?

Answer: [Yes]

Justification: Yes, the claims made in the abstract and intro are supported by sample complexity bounds and experiments.

Guidelines:

- The answer NA means that the abstract and introduction do not include the claims made in the paper.
- The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
- The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
- It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

### 2. Limitations

Question: Does the paper discuss the limitations of the work performed by the authors?

Answer: [Yes]

Justification: Yes, the limitations are discussed in Appendix F where we also present possible ideas to overcome such limitations.

Guidelines:

- The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
- The authors are encouraged to create a separate "Limitations" section in their paper.
- The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
- The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
- The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
- The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
- If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
- While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

### 3. Theory assumptions and proofs

Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [\[Yes\]](#)

Justification: Yes, the main proof idea is clearly explained in the main text in the Analysis section and full proofs are given in the Appendix.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

### 4. Experimental result reproducibility

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [\[Yes\]](#)

Justification: Yes, the proofs are explained to the best of our clarity and should be easy to follow for researchers in the field. Also the experiments are explained in enough detail to be reproducible.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general, releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  - (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  - (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  - (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  - (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

## 5. Open access to data and code

Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Yes, the code is added in the supplementary material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so “No” is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (<https://nips.cc/public/guides/CodeSubmissionPolicy>) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.

- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

## 6. Experimental setting/details

Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

Answer: [\[Yes\]](#)

Justification: Yes, all environments are standard and known in the RL community. So the experiments should be understandable by RL researcher.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
- The full details can be provided either with the code, in appendix, or as supplemental material.

## 7. Experiment statistical significance

Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

Answer: [\[Yes\]](#)

Justification: Yes, we include results averaged over 10 seeds.

Guidelines:

- The answer NA means that the paper does not include experiments.
- The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

## 8. Experiments compute resources

Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

Answer: [\[NA\]](#)

Justification: Our experiments are small scale and can be run on a laptop in within 1/2 days.

Guidelines:

- The answer NA means that the paper does not include experiments.



- The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
- The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
- The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

## 9. Code of ethics

Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics <https://neurips.cc/public/EthicsGuidelines?>

Answer: [Yes]

Justification: Yes, this work is aligned with the NeurIPS Ethics Guidelines.

Guidelines:

- The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
- If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
- The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

## 10. Broader impacts

Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

Answer: [NA]

Justification: This work is fundamental. We do not expect direct impact on the society.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

## 11. Safeguards

Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

Answer: [NA]

Justification: NA



Guidelines:

- The answer NA means that the paper poses no such risks.
- Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
- Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
- We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

## 12. Licenses for existing assets

Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

Answer: [Yes]

Justification: Yes, we cite all relevant works.

Guidelines:

- The answer NA means that the paper does not use existing assets.
- The authors should cite the original paper that produced the code package or dataset.
- The authors should state which version of the asset is used and, if possible, include a URL.
- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, [paperswithcode.com/datasets](https://paperswithcode.com/datasets) has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

## 13. New assets

Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

Answer: [NA]

Justification: NA

Guidelines:

- The answer NA means that the paper does not release new assets.
- Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
- The paper should discuss whether and how consent was obtained from people whose asset is used.
- At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

## 14. Crowdsourcing and research with human subjects

Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

1095 Answer: [NA]  
 1096 Justification: NA  
 1097 Guidelines:

- 1098 • The answer NA means that the paper does not involve crowdsourcing nor research with
- 1099 human subjects.
- 1100 • Including this information in the supplemental material is fine, but if the main contribu-
- 1101 tion of the paper involves human subjects, then as much detail as possible should be
- 1102 included in the main paper.
- 1103 • According to the NeurIPS Code of Ethics, workers involved in data collection, curation,
- 1104 or other labor should be paid at least the minimum wage in the country of the data
- 1105 collector.

1106 **15. Institutional review board (IRB) approvals or equivalent for research with human**  
 1107 **subjects**

1108 Question: Does the paper describe potential risks incurred by study participants, whether  
 1109 such risks were disclosed to the subjects, and whether Institutional Review Board (IRB)  
 1110 approvals (or an equivalent approval/review based on the requirements of your country or  
 1111 institution) were obtained?

1112 Answer: [NA]  
 1113 Justification: NA  
 1114 Guidelines:

- 1115 • The answer NA means that the paper does not involve crowdsourcing nor research with
- 1116 human subjects.
- 1117 • Depending on the country in which research is conducted, IRB approval (or equivalent)
- 1118 may be required for any human subjects research. If you obtained IRB approval, you
- 1119 should clearly state this in the paper.
- 1120 • We recognize that the procedures for this may vary significantly between institutions
- 1121 and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the
- 1122 guidelines for their institution.
- 1123 • For initial submissions, do not include any information that would break anonymity (if
- 1124 applicable), such as the institution conducting the review.

1125 **16. Declaration of LLM usage**

1126 Question: Does the paper describe the usage of LLMs if it is an important, original, or  
 1127 non-standard component of the core methods in this research? Note that if the LLM is used  
 1128 only for writing, editing, or formatting purposes and does not impact the core methodology,  
 1129 scientific rigor, or originality of the research, declaration is not required.

1130 Answer: [NA]  
 1131 Justification: LLMs have not been used.  
 1132 Guidelines:

- 1133 • The answer NA means that the core method development in this research does not
- 1134 involve LLMs as any important, original, or non-standard components.
- 1135 • Please refer to our LLM policy (<https://neurips.cc/Conferences/2025/LLM>)
- 1136 for what should or should not be described.