Inverse Q-Learning Done Right: Offline Imitation Learning in Q^π-Realizable MDPs

Antoine Moulin
Universitat Pompeu Fabra
antoine.moulin@upf.edu

Gergely Neu
Universitat Pompeu Fabra
gergely.neu@gmail.com

Luca Viano
EPFL
luca.viano@epfl.ch

Abstract

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

1 Introduction

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

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

18th European Workshop on Reinforcement Learning (EWRL 2025).

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 Π^{E} that contains the unknown expert policy π_{E}, that is, such that $\pi_{\mathrm{E}} \in \Pi^{\mathrm{E}}$.*

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 $\log |\Pi^{\mathrm{E}}|$ (assuming Π^{E} is finite), meaning large policy classes, potentially necessary to realize the expert, lead to deteriorated guarantees. Additionally, the consequences of misspecification, *i.e.* $\pi_{\mathrm{E}} \notin \Pi^{\mathrm{E}}$, are often severe. For instance, Rohatgi et al. [2025] demonstrated that if the policy class Π^{E} is misspecified, then it is computationally intractable to learn $\arg\min_{\pi \in \Pi^{\mathrm{E}}} \mathcal{D}^2_{\mathsf{H}}(\mathbb{P}^{\pi}, \mathbb{P}^{\pi_{\mathrm{E}}})$, 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 Π^{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^{π}-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 `SPOIL`, standing for Saddle-Point Offline Imitation Learning. In the case of linear function approximation, both update steps of `SPOIL` 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 `SPOIL` 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 $\mathcal{D}_{\mathsf{KL}}(p, q) = \sum_{z \in \mathcal{Z}} p(z) \log \frac{p(z)}{q(z)}$. 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 to make the difference of unnormalized returns of order $\mathcal{O}(\varepsilon)$. We denoted the class of deterministic linear experts as $\Pi^{\mathrm{E}}_{\mathrm{det,lin}} = \left\{\pi : \exists\theta \in \mathfrak{B}(B_\theta), \pi(\cdot) = \arg\max_{a\in\mathcal{A}} \langle\theta, \varphi(\cdot, a)\rangle\right\}$, and an arbitrary policy class as Π^{E}. We also define $W = \max_{\pi\in\Pi^{\mathrm{E}},(x,a)\in\mathcal{X}\times\mathcal{A},h\in[H]} \frac{\pi_{\mathrm{E},h}(a|x)}{\pi_h(a|x)}$ and $\epsilon_{\mathrm{miss}} = \min_{\pi\in\Pi^{\mathrm{E}}} \mathcal{D}^2_{\mathsf{H}}(\mathbb{P}^\pi, \mathbb{P}^{\pi_{\mathrm{E}}})$.

Algorithm	Structural assumptions	Avoids expert realizability	Infinite horizon	Expert class	Expert Traj. (τ_E)
BC with log loss [Foster et al., 2024]	None	✗	✗	Π^{E}	$\mathcal{O}\left(\frac{H^2 \log\lvert\Pi^{\mathrm{E}}\rvert}{\varepsilon^2}\right)$
BC with 0-1 loss [Rajaraman et al., 2021]	None	✗	✗	$\Pi^{\mathrm{E}}_{\mathrm{det,lin}}$	$\widetilde{\mathcal{O}}\left(\frac{H^2 d}{\varepsilon}\right)$
BoostedLogLossBC [Rohatgi et al., 2025]	None	✓ with a misspecification error of $\widetilde{\mathcal{O}}(H\log(W)\epsilon_{\mathrm{miss}})$	✗	Π^{E}	$\mathcal{O}\left(\frac{H^2 \log\lvert\Pi^{\mathrm{E}}\rvert}{\varepsilon^2}\right)$
Projection [Abbeel and Ng, 2004]	Linear reward Known transitions	✓	✓	–	$\widetilde{\mathcal{O}}\left(\frac{d}{(1-\gamma)^2\varepsilon^2}\right)$
MWAL [Syed and Schapire, 2007]	Linear reward Known transitions	✓	✓	–	$\widetilde{\mathcal{O}}\left(\frac{\log(d)}{(1-\gamma)^2\varepsilon^2}\right)$
SPOIL **(Ours)**	Linear Q^π-realizability	✓	✓	–	$\widetilde{\mathcal{O}}\left(\frac{d}{(1-\gamma)^4\varepsilon^2}\right)$
SPOIL **(Ours)**	Q^π-realizability	✓	✓	–	$\widetilde{\mathcal{O}}\left(\frac{\log \mathcal{C}_\epsilon(\mathcal{Q})}{(1-\gamma)^8\varepsilon^4}\right)$

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, $r : \mathcal{X}\times\mathcal{A} \to [0,1]$ is the unknown reward function, $P : \mathcal{X}\times\mathcal{A} \to \Delta(\mathcal{X})$ 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'), $P(x' \mid x, a)$ denotes the probability of landing in state x' after taking action a in state x. A *stationary policy* (or simply *policy*) $\pi : \mathcal{X} \to \Delta(\mathcal{A})$ is a mapping from states to distributions over actions. The interaction of a policy π 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 $A_h \sim \pi(\cdot \mid X_h)$ is taken, a reward $r(X_h, A_h)$ is received, and the agent transitions to a new state $X_{h+1} \sim P(\cdot \mid X_h, A_h)$. 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 π as $V^\pi(x) = \mathbb{E}^\pi\left[\sum_{h=0}^\infty \gamma^h r(X_h, A_h) \,\middle|\, X_0 = x\right]$. Analogously, we define the state-action value function as $Q^\pi(x, a) = \mathbb{E}^\pi\left[\sum_{h=0}^\infty \gamma^h r(X_h, A_h) \,\middle|\, X_0 = x, A_0 = a\right]$. The value functions are tied together via the *Bellman equations*

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

Additionally, we will sometimes use the notation $Q(x, \pi)$ to denote $\sum_a \pi(a|x)Q(x, a)$ for any policy π and any function $Q : \mathcal{X}\times\mathcal{A} \to \mathbb{R}$. Note that this notation allows us to write $V^\pi(x) = Q^\pi(x, \pi)$. Any policy π 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 π. 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")

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

Notably, these definitions and the flow conditions remain valid for general history-dependent policies π that may take the entire history of state-action pairs $(X_1, A_1, \ldots, X_h)$ into account when selecting each action A_h. Finally, we let $\rho^\pi = (1-\gamma)\mathbb{E}^\pi\left[\sum_{h=0}^\infty \gamma^h r(X_h, A_h)\right]$ stand for the total expected normalized return of a (potentially nonstationary) policy π. 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 π be a stationary policy and π' be any policy. Then,*

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

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 $\mathcal{D}^{\pi_{\mathrm{E}}} = \left\{X_E^i, A_E^i\right\}_{i=1}^{\tau_E}$ of state-action pairs sampled from an expert policy's occupancy measure $\mu^{\pi_{\mathrm{E}}}$, our objective is to design an algorithm, Alg, that produces a policy π^{out} satisfying

$$\mathbb{E}\left[\rho^{\pi_{\mathrm{E}}} - \rho^{\pi^{\mathrm{out}}}\right] \le \varepsilon\,. \tag{2}$$

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^π-realizability)**.** *The action value function of any policy π can be written as a linear combination of known features. That is, there exists a known mapping $\varphi : \mathcal{X}\times\mathcal{A} \to \mathbb{R}^d$ such that for any policy π, 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 $\|\theta^\pi\| \le B_\theta$ for all π, and $\sup_{x,a}\|\varphi(x,a)\| \le B_\varphi$.*

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

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

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 ϵ-covering of K if for any $x \in K$, there exists $y \in \mathcal{C}_\epsilon(K,d)$ such that $d(x,y) \le \epsilon$. 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:

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

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:

$$\rho^{\pi_{\mathrm{E}}} - \rho^\pi = \mathcal{L}(\pi; Q^\pi) \le \sup_{Q\in\mathcal{Q}} \mathcal{L}(\pi; Q).$$

This suggests that a good policy π^{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 π^{out} that satisfies $\sup_{Q\in\mathcal{Q}} \mathcal{L}(\pi^{\mathrm{out}}; Q) \le \varepsilon$, then the above inequality implies that the suboptimality of π^{out} as compared to π_{E} will also be at most ε.

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 $k = 1, 2, \ldots, K$, 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

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

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

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

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

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

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

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

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

Proof. The proof simply follows by noticing

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

where we have used the definitions of Δ and Q_k in the first and second lines, respectively. □

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

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

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

Algorithm 1 SPOIL with linear FA

Input: Number of expert trajectories τ_E, learning rate η, number of iterations K.
Initialize: $\theta_0 = 0$, uniform policy π_0.
For $k = 1, 2, \dots, K$**:**

1. $\pi_k(a \mid x) \propto \pi_{k-1}(a \mid x) e^{\eta \langle \varphi(x,a), \theta_{k-1} \rangle}$.
2. $\widehat{g}_k = \tau_E^{-1} \sum_{i=1}^{\tau_E} \big(\varphi(X_E^i, A_E^i) - \varphi(X_E^i, \pi_k)\big)$.
3. $\theta_k = \underset{\theta : \|\theta\| \le B_\theta}{\arg\max} \langle \theta, \widehat{g}_k \rangle = \dfrac{B_\theta}{\|\widehat{g}_k\|} \widehat{g}_k$.

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

Algorithm 2 SPOIL with general FA

Input: Number of expert trajectories τ_E, learning rate η, number of iterations K.
Initialize: $Q_0 = 0$, uniform policy π_0.
For $k = 1, 2, \dots, K$**:**

1. $\pi_k(a \mid x) \propto \pi_{k-1}(a \mid x) e^{\eta Q_{k-1}(x,a)}$.
2. $Q_k \in \underset{Q \in \mathcal{Q}}{\arg\max}\, \widehat{\mathcal{L}}(\pi_k, Q)$.

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 π 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 $\pi_k(a \mid x) \propto e^{\eta \sum_{i=1}^{k-1} Q_i(x,a)}$, 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 $\widehat{\mathcal{L}}$ can be rewritten in terms of the gap between the feature-expectation vectors

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

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

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

and the critic update can be simply written as $\theta_k = \arg\max_{\theta \in \mathfrak{B}(B_\theta)} \langle \theta, \widehat{g}_k \rangle$, 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 1. *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 $\tau_E = \mathcal{O}\Big(\frac{d}{(1-\gamma)^2\varepsilon^2} \log\Big(\frac{B_\theta B_\varphi \log|\mathcal{A}|}{(1-\gamma)\varepsilon}\Big)\Big)$ samples collected by any expert policy π_{E}. Then, the output satisfies $\mathbb{E}\big[\rho^{\pi_{\text{E}}} - \rho^{\pi^{\text{out}}}\big] \le 5\varepsilon$.*

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 Π^{E} realizing the expert policy and as a consequence the bound on τ_E does not scale at all with a complexity measure of Π^{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 $\log|\Pi^{\text{E}}|$ (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^π-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 2. *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 $\tau_E = \mathcal{O}\Big(\frac{\log|\mathcal{A}|}{(1-\gamma)^4\varepsilon^4} \log\Big(\frac{\mathcal{N}_{\epsilon'}(\mathcal{Q}, \|\cdot\|_\infty)}{\varepsilon(1-\gamma)}\Big)\Big)$ samples collected by any expert policy π_{E}, where $\epsilon' = \frac{(1-\gamma)^3\varepsilon^2}{4\log|\mathcal{A}|}$. Then, the output satisfies $\mathbb{E}\big[\rho^{\pi_{\text{E}}} - \rho^{\pi^{\text{out}}}\big] = \mathcal{O}(\varepsilon)$.*

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 2. *For any k and any state-action pair (x, a), consider the sequence of policies starting with π_1 as the uniform policy and updated as $\pi_{k+1}(a \mid x) \propto \pi_k(a \mid x) e^{\eta Q_k(x,a)}$ for some function $Q_k \colon \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ 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 3. *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*

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

We defer the proof to Appendix B. We can use the above result to sketch the proof of Theorem 1.

Proof sketch of Theorem 1. Using Lemma 2 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 3 and the fact that $K^{-1}\sum_{k=1}^{K} \Delta(\pi_k)$ is a random random variable bounded by $(1-\gamma)^{-1}$ almost surely, we obtain the following expectation bound which holds for all $\delta > 0$

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

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 $\frac{2}{K}\sum_{k=1}^{K} \mathbb{E}\left[\Delta_k\right] \leq 4\varepsilon$. Invoking Proposition 1, we conclude that $\mathbb{E}\left[\rho^{\pi_{\mathrm{E}}} - \rho^{\pi^{\mathrm{out}}}\right] \leq 5\epsilon$. The full proof is in Appendix B. □

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 3. 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 4. *Let $\Pi_{\mathcal{Q}}$ 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*

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

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 7). 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 2.

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

5 Numerical experiments

We run experiments to verify that we can imitate efficiently complex experts in linear Q^π 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^π-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 $\Pi_{\mathrm{lin}}^E = \left\{\pi(a|x) = \frac{\exp(\langle\varphi(x,a),\theta\rangle)}{\sum_{b\in\mathcal{A}}\exp(\langle\varphi(x,b),\theta\rangle)}, \theta \in \mathbb{R}^d, \|\theta\| \leq B_\theta\right\}$. In addition, we consider a second expert belonging to the class of three-layer neural networks denoted by Π_{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

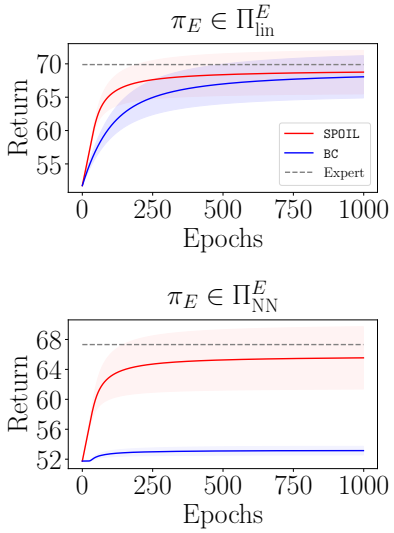


Figure 1: Experiments with simple and complex experts. Curves are averaged across 10 seeds.

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

5.1 Continuous states experiments

We run the general function approximation version of our algorithm in continuous states environmnets from the gym library. In particular, we consider the environments `CartPole-v1`, `Acrobot-v1` and `LunarLander-v2` where the expert is trained via `Soft DQN`. In particular, we use the expert data provided in the code base of Garg et al. [2021]. The learner aims at imitating the expert performance given as input a variable number of expert trajectories. In order to make the task more challenging the trajectories are subsampled each 20 steps in `CartPole-v1`, `Acrobot-v1` and each 5 in `LunarLander-v2`.[1] 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

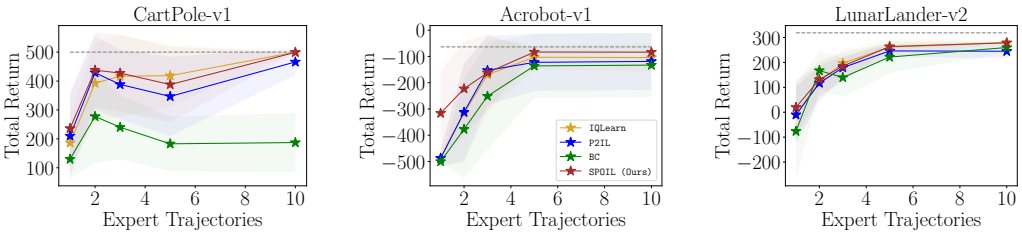


Figure 2: Experiments in continuous states domains. Curves are averaged across 10 seeds.

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

6 Conclusions

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

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

[1] This is common practice in IL experiments (see, *e.g.*, Garg et al., 2021).

Acknowledgments

This project has received funding from the European Research Council (ERC), under the European Union's Horizon 2020 research and innovation programme (Grant agreement No. 950180). This work is funded (in part) through a PhD fellowship of the Swiss Data Science Center, a joint venture between EPFL and ETH Zurich.

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 Lacra 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. Bellemare, Alex Graves, Martin Riedmiller, Andreas K. Fidjeland, Georg Ostrovski, Stig Petersen, Charles Beattie, Amir Sadik, Ioannis Antonoglou, Helen King, Dharshan 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 imitation 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^π-realizability and concentrability. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024. URL https://openreview.net/forum?id=TusuJSbRxm.

Luca Viano, Angeliki Kamoutsi, Gergely Neu, Igor Krawczuk, and Volkan Cevher. Proximal point imitation learning. *Advances in Neural Information Processing Systems*, 35:24309–24326, 2022.

Luca Viano, Stratis Skoulakis, and Volkan Cevher. Imitation learning in discounted linear MDPs without exploration assumptions. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=DChQpB4AJy.

Gellért Weisz, András György, and Csaba Szepesvári. Online rl in linearly q^π-realizable mdps is as easy as in linear mdps if you learn what to ignore. *Advances in Neural Information Processing Systems*, 36:59172–59205, 2023.

Tian Xu, Ziniu Li, Yang Yu, and Zhi-Quan Luo. Provably efficient adversarial imitation learning with unknown transitions. *arXiv preprint arXiv:2306.06563*, 2023.

Contents of Appendix

A Additional related works **13**

B Omitted proofs **15**

B.1 Proof of Lemma 1 (performance difference lemma) 15

B.2 Proof of Lemma 2 (regret of the policy player) 15

B.3 General concentration argument 16

B.4 Proof of Lemma 3 (concentration linear case) 18

B.5 Proof of Theorem 1 (sample complexity guarantee for linear Q^π-realizable MDPs) 20

B.6 Proof of Lemma 4 (concentration nonlinear case) 20

B.7 Proof of Theorem 2 (sample complexity guarantee for Q^π-realizable MDPs) 22

C Technical tools **23**

D On the guarantees of misspecified BC in linear Q^π-realizable MDPs **24**

E Experimental details **24**

F Future directions **25**

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 Π^{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 $\Pi_{\mathrm{det,lin}}$ (defined in the caption of Table 1). They prove a bound on the required number of expert samples of order $\widetilde{\mathcal{O}}\big(\big(H^2 d\big)/\varepsilon\big)$, where d is the feature dimension in the definition of $\Pi_{\mathrm{det,lin}}$. 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, $\pi_{\mathrm{E}} \in \Pi^{\mathrm{E}}$, they showed that one can learn an ε-optimal policy (as defined in Equation (2)) after observing $\mathcal{O}\big(\big(H^2 \log|\Pi^{\mathrm{E}}|\big)/\varepsilon\big)$ samples for a deterministic expert or $\mathcal{O}\big(\big(H^2 \log|\Pi^{\mathrm{E}}|\big)/\varepsilon^2\big)$ 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 ε-optimal policy up to an additional approximation error of $H\log(W)\min_{\pi\in\Pi^{\mathrm{E}}}\mathcal{D}^2_{\mathsf{H}}(\mathbb{P}^\pi,\mathbb{P}^{\pi_{\mathrm{E}}})$. In this context, $\mathbb{P}^\pi$ is the trajectory distribution induced by π, W is a density ratio defined as

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

It is worth noting that these guarantees become vacuous when the policy class Π^{E} is such that at least one policy in Π^{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 $\pi_{\mathrm{E},h}(a\,|\,x)>0$), or if the minimum Hellinger distance $\min_{\pi\in\Pi^{\mathrm{E}}}\mathcal{D}^2_{\mathsf{H}}(\mathbb{P}^\pi,\mathbb{P}^{\pi_{\mathrm{E}}})$ is large. Alternatively, Foster et al. [2024] proved a misspecification result where the additional error is $\min_{\pi\in\Pi^{\mathrm{E}}}\chi^2(\mathbb{P}^\pi,\mathbb{P}^{\pi_{\mathrm{E}}})$. This misspecification error is measured by the χ^2 divergence, with a leading coefficient constant in H and

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

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

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

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

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

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

Following a similar line of research that studies imitation learning from a control-theoretic perspective, Block et al. [2023] studied guarantees for generative behavioural cloning, assuming access to a stabilizing policy dubbed a *synthesis oracle*. These policies can be computed exactly if the dynamics are known, an assumption which is not imposed in our work. However, when provided with such 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 1. *Let π be a stationary policy and π' be any policy. Then,*

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

Proof. Consider the Bellman equations for the stationary policy π. 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

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

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 $\rho^{\pi'} = \sum_{x,a} \mu^{\pi'}(x,a) r(x,a)$. □

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

Next, we apply Lemma 10 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 2. *For any k and any state-action pair (x,a), consider the sequence of policies starting with π_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 $Q_k \colon \mathcal{X}\times\mathcal{A} \to \mathbb{R}$ such that $\|Q_k\|_\infty \le \frac{1}{1-\gamma}$. Then, $\sum_{k=1}^{K} \mathcal{L}(\pi_k, Q_k) \le \frac{\log|\mathcal{A}|}{\eta} + \frac{\eta K}{2(1-\gamma)^2}$.*

Proof. Let us recall that

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

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

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

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

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

It remains to prove the following bound.

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

The result is proven as a particular case of Lemma 10. Specifically, we have that when V is the $|\mathcal{A}|$-dimensional simplex and the Bregman divergence is the KL divergence, it holds that

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

where $\odot$ is the elementwise product. We apply Lemma 10 for each state $x\in\mathcal{X}$, replacing $x_k=\pi_k(\cdot\,|\,x)$ and $\ell_k=-Q_k(x,\cdot)$. We obtain that for the update $\pi_{k+1}(a\,|\,x)\propto\pi_k(a\,|\,x)e^{\eta Q_k(x,a)}$, the guarantee in Lemma 10 holds. Moreover, in this setting we have $\lambda=1$, and $\ell_{\max}=\frac{1}{1-\gamma}$. Given that for any state-action pair (x,a), the initial policy is $\pi_1(a\,|\,x)=\frac{1}{|\mathcal{A}|}$, we have that $D(\pi(\cdot\,|\,x),\pi_1(\cdot\,|\,x))\le\log|\mathcal{A}|$. Thus, we have the following bound

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

and the conclusion follows from $\nu^{\bar{\pi}_{\mathrm{E}}}$ being a probability distribution. □

B.3 General concentration argument

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

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

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

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

The result is the following.

Lemma 5. *Let us consider a value function class $\mathcal{Q}\subset\mathbb{R}^{\mathcal{X}\times\mathcal{A}}$ and the sequences of estimated objective functions $\left\{\widehat{\mathcal{L}}(\pi_k,Q)\right\}_{k=1}^{K}$ for a policy sequence $\{\pi_k\}_{k=1}^{K}$ belonging to a policy class Π. For any $k\in[K]$, recall that for any policy π and function Q, the objective function is defined as*

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

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

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

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

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

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

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

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

where $i \in [\tau_E]$. Each W_i an unbiased estimator of $\mathcal{L}(\pi, Q)$ since π is fixed (*i.e.*, π is not a random quantity depending on the expert data) and $\big(X_E^i, A_E^i\big) \sim \mu^{\pi_{\mathrm{E}}}$ for all $i \in [\tau_E]$. Thus, for any $i \in [\tau_E]$, $\mathbb{E}[W_i] = \mathcal{L}(\pi, Q)$. Moreover, notice that for all $i \in [\tau_E]$, $-\frac{1}{\tau_E(1-\gamma)} \le W_i \le \frac{1}{\tau_E(1-\gamma)}$. Therefore, by an application of Hoeffding's inequality (see Lemma 9), we have that for all $t > 0$,

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

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

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

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

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

Recall that $\mathcal{C}_\epsilon\Big(\mathcal{Q} \times \Pi, \|\cdot\|_{\infty,1}\Big)$ is assumed to be an ϵ-covering set of the space $\mathcal{Q} \times \Pi$ with 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 $(Q, \pi_k) \in \mathcal{Q} \times \Pi$, let $(Q_\epsilon, \pi_{k,\epsilon}) \in \mathcal{C}_\epsilon\Big(\mathcal{Q} \times \Pi, \|\cdot\|_{\infty,1}\Big)$ denote the element of the covering such that $\|(Q, \pi_k) - (Q_\epsilon, \pi_{k,\epsilon})\|_{\infty,1} \le \epsilon$. Then, we have that

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

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

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

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

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

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

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

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

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

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

B.4 Proof of Lemma 3 (concentration linear case)

We now instantiate Lemma 5 in the linear Q^π-realizable setting. For this purpose, we compute a bound on the covering number of the class Π_{lin}, defined in Equation (3).

Lemma 6 (Covering number of Π_{lin})**.** *It holds that the covering number of the policy class Π_{lin} can be bounded as*

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

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

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

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

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

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

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

and

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

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 Cauchy-Schwartz's inequality, we have

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

By 1-Lipshitzness of the softmax function (Lemma 11), it holds that

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

Therefore, the ϵ-covering number for Π_{lin} with respect to the norm $\|\cdot\|_1$, $\mathcal{N}_\epsilon(\Pi_{\text{lin}}, \|\cdot\|_1)$, is upper-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 $\|\cdot\|$, and

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

where we used Lemma 12 in the last inequality. For the second part of the lemma, let us consider $Q, Q' \in \mathcal{Q}_{\text{lin}}$. By definition of $\mathcal{Q}_{\text{lin}}$, there exists $\theta, \theta' \in \mathfrak{B}(B_\theta)$ such that for any state-action pair (x,a), $Q(x,a) = \langle \varphi(x,a), \theta\rangle$ and $Q'(x,a) = \langle \varphi(x,a), \theta'\rangle$. Then,

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

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

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

Finally, the proof is concluded by noting that

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

□

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

Lemma 3. *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*

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

Proof. By Lemma 5, it holds that

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

where the third inequality follows from Lemma 6. □

B.5 Proof of Theorem 1 (sample complexity guarantee for linear Q^π-realizable MDPs)

Theorem 1. *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 π_{E}. Then, the output satisfies $\mathbb{E}\left[\rho^{\pi_{\text{E}}} - \rho^{\pi^{\text{out}}}\right] \leq 5\varepsilon$.*

Proof. By Proposition 1, we have

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

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

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

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 3 and the fact that $\frac{1}{K}\sum_{k=1}^{K} \Delta(\pi_k)$ is a random random variable bounded by $2(1-\gamma)^{-1}$ almost surely, we obtain the following expectation bound which holds for all $\delta > 0$,

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

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

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

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

B.6 Proof of Lemma 4 (concentration nonlinear case)

Before presenting the proof of Theorem 2, we provide a bound on the covering number of the class $\mathcal{Q} \times \Pi_{\mathcal{Q}}$, where $\Pi_{\mathcal{Q}}$ is defined in Equation (4). It turns out that the covering number of this class is exponential in K. In the linear case, the exponential dependence in K was avoided because the state-action value class is closed under addition.

Lemma 7 (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,

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

Proof. Let us consider two policies π and π' 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 $(x, a) \in \mathcal{X} \times \mathcal{A}$, π and π' can be written as

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

and

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

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 11), 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 ϵ-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 ϵ decreases), and therefore, we can write

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

□

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

Lemma 8. *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*

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

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 5, we have that with probability at least $1-\delta$, for any $k \in [K]$, it holds that

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

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

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

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

□

B.7 Proof of Theorem 2 (sample complexity guarantee for Q^π-realizable MDPs)

We are now ready for the proof of Theorem 2, which we restate for convenience.

Theorem 2. *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 $\tau_E = \mathcal{O}\Big(\frac{\log|\mathcal{A}|}{(1-\gamma)^4\varepsilon^4}\log\Big(\frac{\mathcal{N}_{\epsilon'}(\mathcal{Q},\|\cdot\|_\infty)}{\varepsilon(1-\gamma)}\Big)\Big)$ samples collected by any expert policy π_{E}, where $\epsilon' = \frac{(1-\gamma)^3\varepsilon^2}{4\log|\mathcal{A}|}$. Then, the output satisfies $\mathbb{E}\big[\rho^{\pi_{\mathrm{E}}} - \rho^{\pi^{\mathrm{out}}}\big] = \mathcal{O}(\varepsilon)$.*

Proof. Recall that by Proposition 1, we have

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

Then, by Lemma 2, 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 8, with probability at least $1-\delta$, it holds that

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

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 $\delta > 0$

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

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

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

where we denoted $\epsilon' = \frac{1}{\sqrt{2}8K^{3/2}\sqrt{\log(A)A}}$. Setting $\delta = \frac{(1-\gamma)\varepsilon}{4}$ and $K = \frac{2\log A}{(1-\gamma)^2\varepsilon^2}$, we further have

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

Finally, setting

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

we guarantee that

$$\mathbb{E}\Big[\rho^{\pi_{\mathrm{E}}} - \rho^{\pi^{\mathrm{out}}}\Big] = \mathcal{O}(\varepsilon)\,.$$

□

C Technical tools

Lemma 9 (Hoeffding's inequality)**.** *Let $X_1, \ldots, X_n$ be independent random variables such that $|X_i| \leq M$ for all i. Then, for any $\epsilon > 0$,*

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

Lemma 10 (Simplified version of Orabona, 2023, Theorem 6.10)**.** *Let us consider a non-empty closed convex set V, an arbitrary sequence of adaptively chosen loss vectors $(\ell_k)_{k=1}^K$ such that $\|\ell_k\|_\infty \leq \ell_{\max}$, and let $D : V \times \mathrm{int}(V) \to \mathbb{R}$ be a Bregman divergence induced by a λ-strongly convex function in the ℓ_1-norm. Then, for all $u \in V$, the sequence $(x_k)_{k=1}^K$ generated for any k as*

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

for an arbitrary initial x_1 satisfies

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

Lemma 11 (Gao and Pavel, 2017, Proposition 4)**.** *For any $\eta > 0$, let the softmax function be defined for any $z \in \mathbb{R}^n$ as*

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

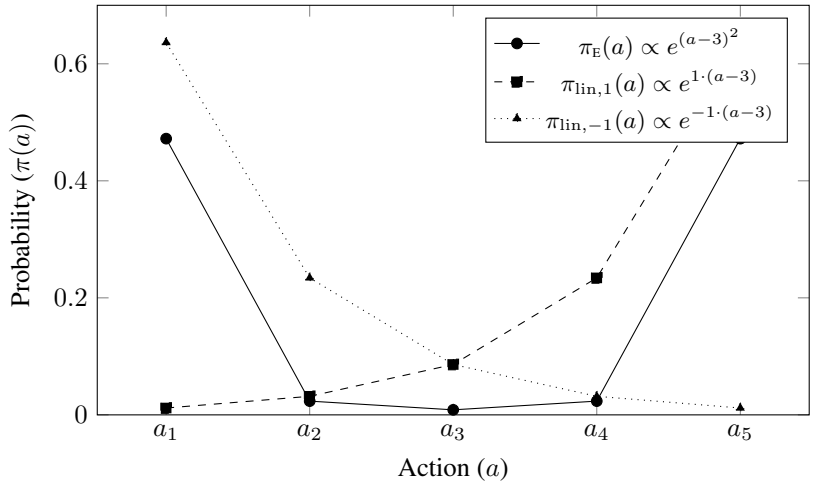
Then, the softmax function is η-Lipschitz with respect to $\|\cdot\|_2$. That is, for any $z, z' \in \mathbb{R}^n$, we have

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

Lemma 12 (Covering number of a Euclidean ball)**.** *The covering number of the Euclidean ball of 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^π-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^π-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 Π_{lin}), can be large even in a simple linear Q^π-realizable MDP.

Consider a single-state MDP defined as follows. Let $A \in \mathbb{N}^*$ be the number of actions, with the action space $\mathcal{A} = [\![1, \ldots, A]\!]$. For each action $a \in \mathcal{A}$, there is a scalar feature $\varphi(a) = -\frac{|\mathcal{A}|}{2} + a \in \mathbb{R}$. To ensure the MDP is linear Q^π-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 π_{E} as

$$\pi_{\mathrm{E}}(a) = \frac{\exp\Big(\varphi(a)^2\Big)}{\sum_{b\in\mathcal{A}} \exp\Big(\varphi(b)^2\Big)}.$$

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 Π_{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-likehood. 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 `P²IL` 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 5×10^{-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`, `P²IL`, and `SPOIL`), 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^π-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 $\{\widehat{g}_k\}_{k=1}^K$. 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 $\left\{\varphi\left(X_E^i, A_E^i\right)\right\}_{i=1}^{\tau_E}$ 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^π realizable MDP.