Bellman-consistent Pessimism for Offline Reinforcement Learning

Anonymous Author(s)
Affiliation
Address
email

Abstract

The use of pessimism, when reasoning about datasets lacking exhaustive exploration, has recently gained prominence in offline reinforcement learning. Despite the robustness it adds to the algorithm, overly pessimistic reasoning can be equally damaging in precluding the discovery of good policies, which is an issue for the popular bonus-based pessimism. In this paper, we introduce the notion of Bellman-consistent pessimism for general function approximation: instead of calculating a point-wise lower bound for the value function, we implement pessimism at the initial state over the set of functions consistent with the Bellman equations. Our theoretical guarantees only require Bellman closedness as standard in the exploratory setting, in which case bonus-based pessimism fails to provide guarantees. Even in the special case of linear MDPs where stronger function-approximation assumptions hold, our result improves upon a recent bonus-based approach by $\mathcal{O}(d)$ in its sample complexity when the action space is finite. Remarkably, our algorithms automatically adapt to the best bias-variance tradeoff in the hindsight, whereas most prior approaches require tuning extra hyperparameters a priori.

1 Introduction

Using past experiences to learn improved behavior for future interactions is a critical capability for a Reinforcement Learning (RL) agent. However, robustly extrapolating knowledge from a historical dataset for sequential decision making is highly challenging, particularly in settings where function approximation is employed to generalize across related observations. In this paper, we provide a systematic treatment of such scenarios with *general* function approximation, and devise algorithms that can provably leverage an arbitrary historical dataset to discover the policy that obtains the largest guaranteed rewards, amongst all possible scenarios consistent with the dataset.

The problem of learning a good policy from historical datasets, typically called batch or offline RL, has a long history [see e.g. 1–3, and references therein]. Many prior works [e.g. 1, 2, 4] make the so-called *coverage assumptions* on the dataset, requiring the dataset to contain any possible state, action pair or trajectory with a lower bounded probability. These assumptions are evidently prohibitive in practice, particularly for problems with large state and/or action spaces. Furthermore, the methods developed under these assumptions routinely display unstable behaviors such as lack of convergence or error amplification, when coverage assumptions are violated [5, 6].

Driven by these instabilities, a growing body of recent literature has pursued a so-called *best effort* style of guarantee instead. The key idea is to replace the stringent assumptions on the dataset with a dataset-dependent performance bound, which gracefully degrades from guaranteeing a near-optimal policy under standard coverage assumptions to offering no improvement over the data collection policy in the most degenerate case. Algorithmically, these works all leverage the principle of *pessimistic extrapolation* from offline data and aim to maximize the rewards the trained agent would

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

obtain in the worst possible MDP that is consistent with the observed dataset. These methods have been shown to be typically more robust to the violation of coverage assumptions in practice, and their theoretical guarantees often provide non-trivial conclusions in settings where the previous results did not apply.

Even though many such best-effort methods have now been developed, very few works provide a comprehensive theory for using generic function approximation, unlike the setting where the dataset satisfies the coverage assumptions [2, 7–10, 4, 11]. For example, [12] provides a partial theory under the assumption of an uncertainty quantification oracle, which however is highly nontrivial to obtain for general function approximation. [13, 14] develop sound theoretical arguments in the tabular setting, which were only heuristically extended to the function approximation setting. The works that explicitly consider function approximation in their design either use an ad-hoc truncation of Bellman backups [15] or strongly rely on particular parameterizations such as linear function approximation [16]. In particular, [15] additionally requires the ability to approximate stationary distribution of the behavior policy, which is a challenging density estimation problem for complex state spaces and cannot be provably performed in the standard linear MDP setting (see Section 3.1).

Our paper takes an important step in this direction. We provide a systematic way to encode pessimism compatible with an arbitrary function approximation class and MDP and give strong theoretical guarantees *without* requiring any coverage assumptions on the dataset. Our first contribution is an information theoretic algorithm that returns a policy with a small regret to any comparator policy, for which coverage assumptions (approximately) hold with respect to the data collection policy. This regret bound is identical to what can be typically obtained when the coverage assumptions hold *for all policies* [2, 4]. But our algorithm requires neither the coverage assumptions, nor additional assumptions such as reliable density estimation for the data generating distribution used by existing best-effort approaches [15]. We furthermore instantiate these results in the special case of linear parameterization; under the linear MDP assumption, our sample complexity bound leads to a factor of $\mathcal{O}(d)$ improvement for a d-dimensional linear MDP, compared with the best known result translated to our discounted setting [16], when the action set is small in size. In addition to the information theoretic algorithm, we also develop a computationally practical version of our algorithm using a Lagrangian relaxation combined with recent advances in soft policy iteration [17–19]. We show that this algorithm can be executed efficiently by querying a (regularized) loss minimization oracle over the value function class, although it has slightly worse theoretical guarantees than the information theoretic version. Both our algorithms display an adaptive property in selecting the best possible form of a bias-variance decomposition, where most prior approaches had to commit to a particular point through their choice of hyperparameters (see the discussion following Theorem 3.1).

2 Preliminaries

Markov Decision Processes We consider dynamical systems modeled as Markov Decision Processes (MDPs). An MDP is specified by $(\mathcal{S}, \mathcal{A}, P, R, \gamma, s_0)$, where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition function with $\Delta(\cdot)$ being the probability simplex, $R : \mathcal{S} \times \mathcal{A} \to [0, R_{\max}]$ is the reward function, $\gamma \in [0, 1)$ is the discount factor, and s_0 is a deterministic initial state, which is without loss of generality. We assume the state and the action spaces are finite but can be arbitrarily large. A (stochastic) policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ specifies a decision-making strategy, and induces a random trajectory $s_0, a_0, r_0, s_1, a_1, r_1, \ldots$, where $a_t \sim \pi(\cdot|s_t)$, $r_t = R(s_t, a_t)$, $s_{t+1} \sim P(\cdot|s_t, a_t)$, $\forall t \geq 0$. We denote the expected discounted return of a policy π as $J(\pi) := \mathbb{E}[\sum_{t=0}^{\infty} \gamma^t r_t|\pi]$, and the learning goal is to find the maximizer of this value: $\pi^\star := \operatorname{argmax}_\pi J(\pi)$. A related concept is the *policy-specific* Q-function, $Q^\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. $Q^\pi(s, a)$ is the discounted return when the trajectory starts with (s, a) and all remaining actions are taken according to π. Q^π is the unique fixed point of the (policy-specific) Bellman operator $\mathcal{T}^\pi : \mathbb{R}^{\mathcal{S}\times\mathcal{A}} \to \mathbb{R}^{\mathcal{S}\times\mathcal{A}}$, defined as:

$$\forall f, \quad (\mathcal{T}^\pi f)(s, a) = R(s, a) + \gamma \mathbb{E}_{s' \sim P(\cdot|s,a)}[f(s', \pi)],$$

where $f(s', \pi)$ is a shorthand for $\mathbb{E}_{a' \sim \pi(\cdot|s')}[f(s', a')]$.

Another important concept is the notion of discounted state-action occupancy, $d_\pi \in \Delta(\mathcal{S} \times \mathcal{A})$, defined as $d_\pi(s, a) := (1 - \gamma)\mathbb{E}[\sum_{t=0}^{\infty} \gamma^t \mathbb{1}[s_t = s, a_t = a]|\pi]$, which characterizes the states and actions visited by a policy π.

Offline RL In the offline setting, the learner only has access to a pre-collected dataset and cannot directly interact with the environment. We assume the standard i.i.d. data generation protocol in our theoretical derivations, that the offline dataset $\mathcal{D}$ consists of n i.i.d. (s, a, r, s') tuples generated as $(s, a) \sim \mu, r = R(s, a), s' \sim P(\cdot|s, a)$ for some *data distribution* μ. We will also use $\mathbb{E}_\mu[\cdot]$ for taking expectation with respect to μ. We will frequently use the data-weighted 2-norm (squared) $\|f\|_{2,\mu}^2 := \mathbb{E}_\mu[f^2]$, and the definition extends when we replace μ with any other state-action distribution ν. The empirical approximation of $\|f\|_{2,\mu}^2$ is $\|f\|_{2,\mathcal{D}}^2 := \frac{1}{n}\sum_{(s,a,r,s')\in\mathcal{D}} f(s, a)^2$.

Function Approximation Function approximation is crucial to generalizing over large and complex state and action spaces. In this work, we search for a good policy in a policy class $\Pi \subset (\mathcal{S} \to \Delta(\mathcal{A}))$ with the help of a value-function class $\mathcal{F} \subset (\mathcal{S} \times \mathcal{A} \to [0, V_{\max}])$ to model Q^π, where $V_{\max} = R_{\max}/(1-\gamma)$. Such a combination is commonly found in approximate policy iteration and actor-critic algorithms [e.g., 20, 21]. For most part of the paper we do not make any structural assumptions on Π and $\mathcal{F}$, making our approach and guarantees applicable to generic function approximators. For simplicity we will assume that these function classes are finite but exponentially large, and use log-cardinality to measure their statistical complexities in the generic results (Section 3 and Section 4). These guarantees easily extend to continuous function classes where log-cardinalities are replaced by the appropriate notions of covering numbers, which we demonstrate when we instantiate our results in the linear MDP setting and work with continuous linear classes (Section 3.1).

We now recall two standard expressivity assumptions on $\mathcal{F}$ [e.g., 2]. To our knowledge, no existing works on offline RL with insufficient data coverage have provided guarantees under these standard assumptions for general function approximation, and they often require stronger or tweaked assumptions (see Section 1).

Assumption 1 (Realizability)**.** *For any $\pi \in \Pi$, we have*

$$\inf_{f\in\mathcal{F}} \sup_{\text{admissible } \nu} \|f - \mathcal{T}^\pi f\|_{2,\nu}^2 \le \varepsilon_{\mathcal{F}},$$

where an admissible distribution ν means that $\nu \in \{d_{\pi'} : \pi' \in \Pi\}$.

Assumption 1 requires that for every $\pi \in \Pi$, there exists $f \in \mathcal{F}$ that well-approximates Q^π. This assumption is often called *realizability*.[1] Technically this is asserted by requiring f to have small Bellman error w.r.t. $\mathcal{T}^\pi$ under all possible *admissible* distributions. As a sufficient condition, we have $\varepsilon_{\mathcal{F}} = 0$ if $Q^\pi \in \mathcal{F}, \forall \pi \in \Pi$.

Assumption 2 (Completeness)**.** *For any $\pi \in \Pi$, we have*

$$\sup_{f\in\mathcal{F}} \inf_{f'\in\mathcal{F}} \|f' - \mathcal{T}^\pi f\|_{2,\mu}^2 \le \varepsilon_{\mathcal{F},\mathcal{F}}.$$

Assumption 2 asserts that $\mathcal{F}$ is approximately closed under $\mathcal{T}^\pi$.[2] Such an assumption is widely used in RL theory and can be only avoided in some rare cases [22]. We also emphasize that we only measure the violation of completeness under μ and do not need to reason about all admissible distributions.

Distribution shift A unique challenge in RL is that the learned policy may induce a state (and action) distribution that is different from the data distribution μ, and the issue is particularly salient when we do not impose any coverage assumption on μ. Therefore, it is important to carefully characterize the distribution shift, which we measure using the following definition, which generalizes prior definitions specific to linear function approximation [19, 23]:

Definition 1. *We define $\mathscr{C}(\nu; \mu, \mathcal{F}, \pi)$ as follows to measure the distribution shift from an arbitrary distribution ν to the data distribution μ, w.r.t. $\mathcal{F}$ and π,*

$$\mathscr{C}(\nu; \mu, \mathcal{F}, \pi) := \max_{f\in\mathcal{F}} \frac{\|f - \mathcal{T}^\pi f\|_{2,\nu}^2}{\|f - \mathcal{T}^\pi f\|_{2,\mu}^2}.$$

[1] In the exploratory setting, realizability is usually stated in the form of $\inf_{f\in\mathcal{F}} \|f - \mathcal{T}^\pi f\|_{2,\mu}^2 \le \varepsilon_{\mathcal{F}}$ for any $\pi \in \Pi$. However, the exploratory setting also usually has data coverage assumptions in the form of $\sup_\nu \|\nu/\mu\|_\infty \le C$. Combining them together implies Assumption 1.

[2] Sometimes completeness implies realizability, so the latter does not need to be assumed separately [4]. However, this often relies on μ being exploratory, which is not the case here.

Intuitively, $\mathscr{C}(\nu; \mu, \mathcal{F}, \pi)$ measures how well Bellman errors under π transfer between the distributions ν and μ. For instance, a small value of $\mathscr{C}(d_\pi; \mu, \mathcal{F}, \pi)$ enables accurate policy evaluation for π using data collected under μ. More generally, we observe that

$$\mathscr{C}(\nu; \mu, \mathcal{F}, \pi) \leq \|\nu/\mu\|_\infty := \sup_{s,a} \frac{\nu(s,a)}{\mu(s,a)}, \quad \text{for any } \pi, \mathcal{F}.$$

and the RHS is a classical notion of bounded distribution ratio for error transfer (e.g., [9, 4, 11]). Moreover, our measure can be tighter than $\|\nu/\mu\|_\infty$: Even two distributions ν and μ that are sufficiently disparate might admit a reasonable transfer, so long as this difference is not detected by π and $\mathcal{F}$. To this end, our definition better captures the crucial role of function approximation in generalizing across different states. As an example, in the special case of linear MDPs, full coverage under our definition (i.e., boundedness of $\mathscr{C}$ *for all admissible* ν) can be implied from the standard coverage assumption for linear MDPs that considers the spectrum of the feature covariance matrix under μ; see Section 3.1 for more details.

3 Information-Theoretic Results with Bellman-consistent Pessimism

In this section, we provide our first theoretical result which is information-theoretic, in that it uses a computationally inefficient algorithm. The approach uses the offline dataset to first compute a lower bound on the value of each policy $\pi \in \Pi$, and then returns the policy with the highest pessimistic value estimate. While this high-level template is at the heart of many recent approaches [e.g., 13, 24, 15, 12, 25, 14], our main novelty is in the design and analysis of *Bellman-consistent pessimism* for general function approximation.

For a policy π, we first form a *version space* of all the functions $f \in \mathcal{F}$ which have a small Bellman error under the evaluation operator $\mathcal{T}^\pi$. We then return the predicted value of π in the initial state s_0 by the functions in this version space. The use of pessimism at the initial state, while maintaining Bellman consistency (by virtue of having a small Bellman error) limits over pessimism, which is harder to preclude in the pointwise pessimistic penalties used in some other works [16]. More formally, given a dataset $\mathcal{D}$, let us define

$$\mathcal{L}(f', f, \pi; \mathcal{D}) := \frac{1}{n} \sum_{(s,a,r,s') \in \mathcal{D}} \left(f'(s,a) - r - \gamma f(s', \pi)\right)^2,$$

and an empirical estimate of the Bellman error $\mathcal{E}(f, \pi; \mathcal{D})$ is

$$\mathcal{E}(f, \pi; \mathcal{D}) := \mathcal{L}(f, f, \pi; \mathcal{D}) - \min_{f' \in \mathcal{F}} \mathcal{L}(f', f, \pi; \mathcal{D}). \tag{3.1}$$

Our algorithm. With this notation, our information-theoretic approach finds a policy by optimizing:

$$\widehat{\pi} = \operatorname*{argmax}_{\pi \in \Pi} \min_{f \in \mathcal{F}_{\pi,\varepsilon}} f(s_0, \pi), \quad \text{where } \mathcal{F}_{\pi,\varepsilon} = \{f \in \mathcal{F} : \mathcal{E}(f, \pi; \mathcal{D}) \leq \varepsilon\}, \tag{3.2}$$

In the formulation above, $\mathcal{F}_{\pi,\varepsilon}$ is the version space of policy π. To better understand the intuition behind the estimator in Equation 3.2, let us define

$$f_{\pi,\min} := \operatorname*{argmin}_{f \in \mathcal{F}_{\pi,\varepsilon_r}} f(s_0, \pi), \;\; f_{\pi,\max} := \operatorname*{argmax}_{f \in \mathcal{F}_{\pi,\varepsilon_r}} f(s_0, \pi), \;\; \text{and } \Delta f_\pi(s,a) := f_{\pi,\max}(s,a) - f_{\pi,\min}(s,a).$$

Intuitively, if the parameter ε is defined to ensure that Q^π (or its best approximation in $\mathcal{F}$) is in $\mathcal{F}_{\pi,\varepsilon}$, we easily see that $\Delta f_\pi(s_0, \pi)$ is an upper bound on the error in our estimate of $J(\pi)$ for any $\pi \in \Pi$. In fact, an easy argument in our analysis shows that if $Q^\pi \in \mathcal{F}_{\pi,\varepsilon}$ for all $\pi \in \Pi$, then $\Delta f(s_0, \pi)$ is an upper bound on the regret $J(\pi) - J(\widehat{\pi})$ of our estimator relative to any π we wish to compete with.

Theoretical analysis. To leverage this observation, we first define the a critical threshold ε_r which ensures that (the best approximation of) Q^π is indeed contained in our version spaces for all π:

$$\varepsilon_r := \frac{139 V_{\max}^2 \log \frac{|\mathcal{F}||\Pi|}{\delta}}{n} + 39\varepsilon_{\mathcal{F}}. \tag{3.3}$$

With this definition, we now give a more refined bound on the regret of our algorithm (3.2) by further splitting the error estimate $\Delta f_\pi(s_0, \pi)$ which is random owing to its dependence on the version space, and analyze it through a novel decomposition into on-support and off-support components. While we bound the on-support error using standard techniques, the off-support error is akin to a bias term which captures the interplay between the data collection distribution and function approximation in the quality of the final solution.

Theorem 3.1. *Let $\varepsilon = \varepsilon_r$ where is ε_r defined in Eq.(3.3) and $\widehat{\pi}$ be obtained by Eq.(3.2). Then, for any policy $\pi \in \Pi$ and any constant $C_2 \geq 1$, with probability at least $1-\delta$,*

$$J(\pi) - J(\widehat{\pi}) \leq \underbrace{\mathcal{O}\left(\frac{V_{\max}\sqrt{C_2}}{1-\gamma}\sqrt{\frac{\log\frac{|\mathcal{F}||\Pi|}{\delta}}{n}} + \frac{\sqrt{C_2(\varepsilon_{\mathcal{F},\mathcal{F}} + \varepsilon_{\mathcal{F}})}}{1-\gamma}\right)}_{\mathrm{err}_{\mathrm{on}}(\pi)\text{: on-support error}}$$
$$+ \underbrace{\frac{1}{1-\gamma} \cdot \min_{\nu:\mathscr{C}(\nu;\mu,\mathcal{F},\pi)\leq C_2} \sum_{(s,a)\in\mathcal{S}\times\mathcal{A}} (d_\pi \setminus \nu)(s,a)\left[\Delta f_\pi(s,a) - (\mathcal{T}^\pi \Delta f_\pi)(s,a)\right]}_{\mathrm{err}_{\mathrm{off}}(\pi)\text{: off-support error}},$$

where $\mathscr{C}(\nu;\mu,\mathcal{F},\pi)$ is defined in Definition 1 and $(d_\pi \setminus \nu)(s,a) \coloneqq \max(d_\pi(s,a) - \nu(s,a), 0)$.

Bias-variance decomposition. Note that decomposition of our error bound into on-support and off-support parts effectively achieves a bias-variance tradeoff. A small value of the concentrability threshold C_2 requires the choice of the distribution ν closer to μ, which results in better estimation error guarantee (which is $\mathcal{O}(\sqrt{C_2/n})$) when we transfer from μ to ν, but potentially pays a high bias due to the mismatch between d_π and ν. A larger threshold permits more flexibility in choosing ν similar to d_π for a smaller bias, but results in a larger variance and estimation error. Rather than commit to a particular trade-off, our estimator automatically adapts to the best possible splitting (Figure 1 illustrates this concept) by allowing us to choose the best threshold C_2. The on-support part matches the n rate (fast rate error bound) of API or AVI analysis (e.g, [26, 27, 4]). The dependency on horizon is only linear and matches the best previous result with concentrability assumption [11]. For the off-support part, it depends on the off-support mass $d_\pi \setminus \nu$ and the "quality" of the off-support estimation: if all value functions in the version space are close to each other in the off-support region for policy π, the gap between $J(\pi)$ and $J(\widehat{\pi})$ can still be small even with a large off-support mass. The following corollary formally states this property.

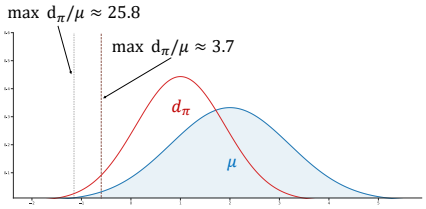

Figure 1: An example illustrating different on-support and off-support splittings (denoted by two different vertical lines). Different splitting has different C_2 values, and further yields different bias-variance trade-offs.

Corollary 1 ("Double Robustness")**.** *Under conditions of Theorem 3.1, for any π and $C_2 \geq 0$, $\mathrm{err}_{\mathrm{off}}(\pi) = 0$ when either (1) $\mathscr{C}(d_\pi;\mu,\mathcal{F},\pi) \leq C_2$, or, (2) $\Delta f_\pi - \mathcal{T}^\pi \Delta f_\pi \equiv 0$.*

Adaptive guarantees by algorithm design. As mentioned above, Theorem 3.1 implicitly selects the best bias-variance decomposition through the best choice of C_2 in hindsight, with this decomposition being purely a proof technique, not an knob in the algorithm. In contrast, many prior approaches [15, 13, 24] employ explicit thresholds to control density ratios in their algorithms, which makes the tradeoff a hyperparameter in their algorithms. Since choosing hyperparameters is particularly challenging in offline RL, where even policy evaluation can be unreliable, this novel axis of adaptivity is an extremely desirable property of our approach.

Comparison to guarantees in the exploratory setting. When a dataset with full coverage is given, classical analyses provide near-optimality guarantees that compete with the optimal policy $\pi^\star$ with a polynomial sample complexity, when $\pi^\star \in \Pi$ and both realizability and completeness hold for $\mathcal{F}$; see Antos et al. [2] for a representative analysis. As mentioned earlier, such analysis often requires boundedness of $\|\nu/\mu\|_\infty$ *for all admissible distributions* $\nu \in \{d_\pi : \pi \in \Pi\}$. On the other hand, it is easily seen that we can compete with $\pi^\star$ under much weaker conditions.

Corollary 2 (Competing with optimal policy)**.** *Under conditions of Theorem 3.1, if* $\mathscr{C}(d_{\pi^\star}; \mu, \mathcal{F}, \pi^\star) \leq C_2$*, we have*

$$J(\pi^\star) - J(\widehat{\pi}) \leq \mathcal{O}\left(\frac{V_{\max}\sqrt{C_2}}{1-\gamma}\sqrt{\frac{\log\frac{|\mathcal{F}||\Pi|}{\delta}}{n}} + \frac{\sqrt{C_2(\varepsilon_{\mathcal{F},\mathcal{F}} + \varepsilon_{\mathcal{F}})}}{1-\gamma}\right).$$

Notably, these milder coverage assumptions in Corollaries 1 and 2 provide offline RL counterparts to the benefits of policy-gradient-style methods with *online* access to the environment [28, 29, 19].

Comparison with Liu et al. [15]. The closest prior result to our work is that of Liu et al. [15], who develop a pessimistic estimator that truncates Bellman backups from state-action pairs infrequently visited by μ, and analyzes the resulting pessimistic policy and value iteration algorithms under general function approximation. For truncating Bellman backups, however, their work requires estimating the state-action distribution of data, which can be challenging in high-dimensions and they incur additional errors from density estimation which we avoid. Further, their algorithms only compete with policies π where $\|d_\pi/\mu\|_\infty$ is bounded instead of the more general result that we provide, and makes their results vacuous in a linear MDP setting under typical feature coverage assumptions.

Safe Policy Improvement. Some prior works [e.g. 30, 15] discuss the scenario where the dataset $\mathcal{D}$ is collected with a behavior policy π_b with $\mu = d_{\pi_b}$, and demonstrate that their algorithms always return a policy competitive with π_b. In our setup, this is straightforward as d_{π_b} is always covered, as shown next.

Corollary 3 (Bounded degradation from behavior policy)**.** *Under conditions of Theorem 3.1, if* $\mu = d_{\pi_b}$ *for some policy* $\pi_b \in \Pi$*, we have*

$$J(\pi_b) - J(\widehat{\pi}) \leq \mathcal{O}\left(\frac{V_{\max}}{1-\gamma}\sqrt{\frac{\log\frac{|\mathcal{F}||\Pi|}{\delta}}{n}} + \frac{\sqrt{\varepsilon_{\mathcal{F},\mathcal{F}} + \varepsilon_{\mathcal{F}}}}{1-\gamma}\right).$$

Proof sketch of Theorem 3.1. We now briefly describe the core ideas in the proof. More detailed arguments are deferred to the full proof in Appendix B.1.

The key to prove Theorem 3.1 is to translate the $J(\pi) - J(\widehat{\pi})$ to the Bellman error of value functions in the version space $\mathcal{F}_{\pi,\varepsilon}$. Our main proving strategies are as follows:

1. As the selection of $\varepsilon = \varepsilon_r$ ensures the accurate estimation Q^π is contained in the version space $\mathcal{F}_{\widehat{\pi},\varepsilon}$ for any π, we can obtain $J(\pi) - J(\widehat{\pi}) \leq \max_{f\in\mathcal{F}_{\pi,\varepsilon}} f(s_0,\pi) - \min_{f\in\mathcal{F}_{\widehat{\pi},\varepsilon}} f(s_0,\widehat{\pi}) +$ approximation error.
2. By the optimality of $\widehat{\pi}$, we have $\min_{f\in\mathcal{F}_{\widehat{\pi},\varepsilon}} f(s_0,\widehat{\pi}) \geq \min_{f\in\mathcal{F}_{\pi,\varepsilon}} f(s_0,\pi)$. This indicates that $J(\pi) - J(\widehat{\pi}) \leq \max_{f\in\mathcal{F}_{\pi,\varepsilon}} f(s_0,\pi) - \min_{f\in\mathcal{F}_{\pi,\varepsilon}} f(s_0,\pi) +$ approximation error.
3. By using a standard telescoping argument (e.g., [11, Lemma 1]), $\max_{f\in\mathcal{F}_{\pi,\varepsilon}} f(s_0,\pi) - \min_{f\in\mathcal{F}_{\pi,\varepsilon}} f(s_0,\pi)$ can be upper bounded by the Bellman error of $\operatorname{argmax}_{f\in\mathcal{F}_{\pi,\varepsilon}} f(s_0,\pi)$ and $\operatorname{argmin}_{f\in\mathcal{F}_{\pi,\varepsilon}} f(s_0,\pi)$ over distribution d_π.

After combining all the three steps above together and considering the distribution shift effect, we complete the proof. □

3.1 Results in Linear MDPs

Here we perform a case study in linear MDPs [31], an important theoretical model in RL that sits in between the tabular setting and generic function approximation. We will show that our results—when instantiated in linear MDPs—automatically provides state-of-the-art guarantees, improving over existing results specialized to this setting by a factor of $\mathcal{O}(d)$ [16] when the action space is finite.

We recall the linear MDP setup (we set $R_{\max} = 1$ and $V_{\max} = \frac{1}{1-\gamma}$ for consistency with literature).

Definition 2 (Linear MDP [31])**.** *Let* $\phi : \mathcal{S}\times\mathcal{A} \to \mathbb{R}^d$ *be a feature mapping. An MDP is called a linear MDP if there exists a feature mapping* $\psi : \mathcal{S} \to \mathbb{R}^d$ *and* $\theta_R \in \mathbb{R}^d$ *with* $\|\phi(s,a)\|_2, \|\theta_R\|_2 \leq 1$*, such that the following holds for any* $(s,a,s') \in \mathcal{S}\times\mathcal{A}\times\mathcal{S}$*,*

$$P(s'|s,a) = \phi(s,a)^\top\psi(s'), \qquad R(s,a) = \phi(s,a)^\top\theta_R.$$

Given feature mapping ϕ, we also define the value-function class $\mathcal{F}_\Phi$ as $\mathcal{F}_\Phi := \{\phi(\cdot,\cdot)^\mathsf{T}\theta : \theta \in \mathbb{R}^d, \phi(\cdot,\cdot)^\mathsf{T}\theta \in [0, V_{\max}]\}$, and the policy class Π_Φ consists of the greedy policies of each value function in $\mathcal{F}_\Phi$. Such choices automatically ensure that $\pi^\star \in \Pi_\Phi$ and $\mathcal{F}_\Phi$ satisfies Assumptions 1 and 2 exactly with $\varepsilon_{\mathcal{F},\mathcal{F}} = \varepsilon_{\mathcal{F}} = 0$ [31]. In contrast, we highlight that the standard linear MDP setup does *not* entail all the assumptions needed by Liu et al. [15] as mentioned earlier.

Below is our main result in the linear MDP setting.

Theorem 3.2. *Suppose MDP $\mathcal{M}$ is a linear MDP (Definition 2) and $\widehat{\pi}$ is the output of Eq.(3.2) using value-function class $\mathcal{F}_\Phi$ and policy class Π_Φ. If we choose $\varepsilon = cV_{\max}^2 d \log \frac{V_{\max}|\mathcal{A}|d}{\delta}/n$, then, for any policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$, we have*

$$J(\pi) - J(\widehat{\pi}) \le \mathcal{O}\left(\frac{V_{\max}}{1-\gamma}\sqrt{\frac{d\log\frac{V_{\max}|\mathcal{A}|d}{\delta}}{n}}\mathbb{E}_{d_\pi}\left[\sqrt{\phi(s,a)^\mathsf{T}\Sigma_{\mathcal{D}}^{-1}\phi(s,a)}\right]\right),$$

where c is an absolute constant, and $\Sigma_{\mathcal{D}} := \mathbb{E}_{\mathcal{D}}\left[\phi(s,a)\phi(s,a)^\mathsf{T}\right]$.

The detailed proof of Theorem 3.2 is provided in Appendix B.2. Our guarantee is structurally very similar to that of Jin et al. [16, Theorem 4.4], except that they consider the finite-horizon setting. If we translate their result to the discounted setting by setting $H = \mathcal{O}(1/(1-\gamma))$, we enjoy a net improvement of order $\mathcal{O}(d)$ in sample complexity when the action space is finite. To make it concrete, that is $\mathcal{O}(\sqrt{d^2\log(dn/\delta)/n})$ vs. $\mathcal{O}(\sqrt{d\log(d|\mathcal{A}|/\delta)/n})$ error bounds. The bound also shows that having a full-rank $\Sigma_{\mathcal{D}}$ (which is ensured by a full-rank covariance under μ) is sufficient for consistent offline RL in linear MDPs. Crucially, the full-rank covariance is an easily checkable condition from data, as opposed to unverifiable concentrability assumptions. As a caveat, our results do not imply a computationally efficient algorithm, as a naïve implementation involves evaluating each policy pessimistically to pick the best. We discuss a computationally efficient adaptation of our approach in the next section.

4 Practical Algorithm — Regularized Offline Policy Optimization

A major challenge using the proposed algorithm in Section 3 in practice is that searching the policy with the best pessimistic evaluation over the policy space Π is not computationally tractable. In this section, we present a practical algorithm that is computationally efficient assuming access to a (regularized) loss minimization oracle over the value function class $\mathcal{F}$, and also comes with rigorous theoretical guarantees.

Algorithm 1 PSPI: Pessimistic Soft Policy Iteration

Input: Batch data $\mathcal{D}$, regularization coefficient λ.

1: Initialize policy π_1 as the uniform policy.
2: **for** $t = 1, 2, \ldots, T$ **do**
3: Obtain the pessimistic estimation for π_t as f_t,

$$f_t \leftarrow \operatorname*{argmin}_{f\in\mathcal{F}} \left(f(s_0,\pi_t) + \lambda\mathcal{E}(f,\pi_t;\mathcal{D})\right),$$

where $\mathcal{E}(f,\pi_t;\mathcal{D})$ is defined in Eq.(3.1).

4: Calculate π_{t+1} by,

$$\pi_{t+1}(a|s) \propto \pi_t(a|s)\exp\left(\eta f_t(s,a)\right),\ \forall s,a \in \mathcal{S}\times\mathcal{A}.$$

5: **end for**
6: Output $\bar{\pi} := \mathsf{Unif}(\pi_{[1:T]})$. ▷ *uniformly mix $\pi_1, \ldots, \pi_T$ at the trajectory level*

Our practical algorithm is summarized in Algorithm 1. It has three key differences from the information-theoretic version in Eq.(3.2):

1. The pessimistic policy evaluation is now performed via regularization (Line 3) instead of constrained optimization.

2. Instead of searching over an explicit policy space Π, we search over a policy class implicitly induced from $\mathcal{F}$ (defined in Equation 4.1) and therefore no longer have a policy class independent of $\mathcal{F}$ separately, which is a common practice in API-style algorithms [7, 2, 27].
3. We optimize the policy using mirror descent updates, which yields computationally tractable optimization over the implicit policy class. This property has been leveraged in many prior works, although typically in online RL settings [17, 32, 33, 19, 18].

Note that the use of a specific policy class above can be relaxed if a stronger structural assumption is made on the MDP (e.g., linear MDPs [31, 16]).

4.1 Analysis of Algorithm 1

We now provide the analysis of Algorithm 1 in this section. For ease of presentation, we formally define the implicit policy class for this section:

$$\Pi_{\text{SPI}} := \{\pi'(\cdot|s) \propto \exp(\eta \sum_{i=1}^{t} f^{(t)}(s,\cdot)) : 1 \le t \le T, f^{(1)}, \dots, f^{(i)} \in \mathcal{F}\},$$

which is the natural policy class for soft policy-iteration approaches. The following theorem describes the performance guarantee of $\bar{\pi}$.

Theorem 4.1. *Let $\lambda = \frac{1}{(1-\gamma)\sqrt{\varepsilon_r}}$ with ε_r in Eq.(3.3), $\eta = \sqrt{\frac{\log|\mathcal{A}|}{2V_{\max}^2 T}}$, and $\bar{\pi}$ be obtained from Algorithm 1. For any policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ we wish to compete with, suppose Assumptions 1 and 2 hold with respect to the policy class $\Pi_{SPI} \cup \{\pi\}$. Then, for any constant $C_2 \ge 1$, we have with probability at least $1-\delta$,*

$$\begin{aligned} & J(\pi) - J(\bar{\pi}) \\ \le\ & \underbrace{\mathcal{O}\left(\sqrt{C_2}\left(\frac{\sqrt{\varepsilon_{\mathcal{F},\mathcal{F}} + \varepsilon_{\mathcal{F}}}}{1-\gamma} + \frac{V_{\max}}{1-\gamma}\sqrt[3]{\frac{T\log\frac{|\mathcal{F}|}{\delta}}{n}} + \sqrt[3]{\frac{V_{\max}\varepsilon_{\mathcal{F}}}{(1-\gamma)^2}}\right)\right)}_{\mathrm{err}_{\mathrm{on}}(\pi)\text{: on-support error}} + \underbrace{\mathcal{O}\left(\frac{V_{\max}}{1-\gamma}\sqrt{\frac{\log|\mathcal{A}|}{T}}\right)}_{\text{optimization error}} \\ & + \underbrace{\frac{1}{T}\sum_{t=1}^{T}\left(\min_{\nu:\mathscr{C}(\nu;\mu,\mathcal{F},\pi_t)\le C_2}\left|\sum_{(s,a)\in\mathcal{S}\times\mathcal{A}}(d_\pi \setminus \nu)(s,a)\left[f_t(s,a) - (\mathcal{T}^{\pi_t} f_t)(s,a)\right]\right|\right)}_{\mathrm{err}_{\mathrm{on}}(\pi)\text{: off-support error}}, \end{aligned}$$

where $\mathscr{C}(\nu;\mu,\mathcal{F},\pi_t)$ is defined in Definition 1, $(d_\pi \setminus \nu)(s,a) := \max(d_\pi(s,a) - \nu(s,a), 0)$.

We provide a proof sketch of Theorem 4.1 at the end of this section, and defer the full proof to Appendix C. We now make a few remarks about the results in Theorem 4.1.

Measurement of distribution shift effect. Compared with the information-theoretical result (provided in Theorem 3.1), the measurement of distribution shift in Theorem 4.1 depends on the optimization trajectory. That is, it measures the distance between two distribution ν and μ by $\mathscr{C}(\nu;\mu,\mathcal{F},\pi_t)$ ($\pi_{[1:T]}$ is the sequence of policies produced by the algorithm) whereas Theorem 3.1 uses $\mathscr{C}(\nu;\mu,\mathcal{F},\pi)$ (π is the baseline policy we compete with). We remark that both of these two measurements are weaker then traditional density-ratio definitions (e.g., [9, 4, 11]) as we demonstrated before, as the dependence of $\mathscr{C}$ on π is relatively secondary.

Dependence on T. The number of optimization rounds T affects the bound in two opposite ways: as T increases, the optimization error term decreases, whereas the second term of the on-support error increases. The latter increase is due to the complexity of the implicit policy class Π growing exponentially with T, which affects our concentration bounds. To optimize the bound, the optimal choice is $T = \mathcal{O}(n^{2/5})$, leading to an overall $\mathcal{O}(n^{-1/5})$ rate. While such a rate is relatively slow, we remark that the complexity bound of Π is conservative, and in certain cases it is possible to obtain much sharper bounds: for example, in linear MDPs (Section 3.1), Π_{SPI} are a priori captured by the space of softmax policies, whose complexity has no dependence on T (up to mild logarithmic dependence due to norms). That is, the $\mathrm{err}_{\mathrm{on}}(\pi)$ term in Theorem 4.1 reduces to $\widetilde{\mathcal{O}}(\frac{V_{\max}}{1-\gamma}\sqrt[3]{d/n})$ ($\varepsilon_{\mathcal{F},\mathcal{F}} = \varepsilon_{\mathcal{F}} = 0$ in linear MDPs), and yields an overall $\mathcal{O}(n^{-1/3})$ rate.

Bias-variance decomposition. Similar to Theorem 3.1, Theorem 4.1 also allows arbitrary decomposition of the error bound into on-support and off-support components by setting the concentrability threshold C_2, which serves as a bias-variance tradeoff as before. In fact, the splitting can be done separately for each π_t in $1 \leq t \leq T$ and we omit such flexibility for readability. The optimization error does not depend on the splitting. Our performance guarantee naturally adapts to the best possible decomposition as before. As in Theorem 3.1, if the estimation on the off-support region is "high-quality", we can further simplify the performance guarantees, but the requirement of "high-quality" is different from that of Corollary 1. We make it formal in the following corollary.

Corollary 4 ("Double Robustness")**.** *For any π and $C_2 \geq 0$, $\mathrm{err}_{\mathrm{off}}(\pi) = 0$ when either (1) $\mathscr{C}(d_\pi; \mu, \mathcal{F}, \pi_t) \leq C_2$ for all $t \in [T]$, or, (2) $f_t - \mathcal{T}^{\pi_t} \Delta f_t \equiv 0$ for all $t \in [T]$.*

We note that the conditions above depend on the optimization trajectory through their dependence on π_t, but can be made algorithm-independent by instead asserting the stronger requirement that $\mathscr{C}(d_\pi, \mu, \mathcal{F}, \pi') \leq C_2$ for all $\pi' \in \Pi_{\text{SPI}}$ in the first condition.

Competing with the optimal policy. As before, we can provide a guarantee for competing with the optimal policy, under coverage assumptions weaker than the typical batch RL literature, albeit slightly stronger than those of Corollary 2. We state the formal result below.

Corollary 5 (Competing with optimal policy)**.** *Under conditions of Theorem 4.1, if $\mathscr{C}(d_{\pi^\star}; \mu, \mathcal{F}, \pi) \leq C_2$ for all $\pi \in \Pi_{SPI}$, we have*

$$J(\pi^\star) - J(\widehat{\pi}) \leq \mathcal{O}\left(\frac{V_{\max}\sqrt{C_2}}{1-\gamma} \left(\frac{\log \frac{|\mathcal{F}|}{\delta} \log |\mathcal{A}|}{n} \right)^{1/5} + \frac{\sqrt{C_2(\varepsilon_{\mathcal{F},\mathcal{F}} + \varepsilon_{\mathcal{F}})}}{1-\gamma} \right).$$

Note that the conditions of Corollary 5 are satisfied as before whenever $\|d_{\pi^\star}/\mu\|_\infty \leq C_2$.

We conclude the section with a proof sketch showing the key insights used in establishing the proof.

Proof sketch of Theorem 4.1. Our proof constructs a corresponding MDP $\mathcal{M}_t$ for every f_t, π_t pair. Each $\mathcal{M}_t$ has the same dynamics as the ground-truth MDP, but chooses a different reward function, such that f_t is the Q-function of π_t in $\mathcal{M}_t$, $Q^{\pi_t}_{\mathcal{M}_t}$ (we use the subscript of $\mathcal{M}_t$ to denote the corresponding value or operator in MDP $\mathcal{M}_t$). Our proof relies on some key properties of $\mathcal{M}_t$, such as $Q^\pi - \mathcal{T}^\pi_{\mathcal{M}_t} Q^\pi = f_t - \mathcal{T}^{\pi_t} f_t$. We decompose $J(\pi) - J(\bar{\pi})$ as follows.

$$J(\pi) - J(\bar{\pi}) \leq \underbrace{\frac{1}{T}\sum_{t=1}^{T} (J_{\mathcal{M}_t}(\pi) - J_{\mathcal{M}_t}(\pi_t))}_{\text{optimization error}} + \underbrace{\frac{1}{T}\sum_{t=1}^{T} (J(\pi) - J_{\mathcal{M}_t}(\pi))}_{\text{controlled by } \|Q^\pi - \mathcal{T}^\pi_{\mathcal{M}_t} Q^\pi\|_{2,d_\pi} = \|f_t - \mathcal{T}^{\pi_t} f_t\|_{2,d_\pi}}$$

$$+ \text{ approximation/statistical errors.}$$

The proof is completed by bounding $\|f_t - \mathcal{T}^{\pi_t} f_t\|_{2,d_\pi}$ on both on-support and off-support regions. □

5 Conclusions

This paper investigates sample-efficient offline reinforcement learning without data coverage assumptions (e.g., concentrability). To achieve that goal, our paper contributes several crucial improvements to the literature. We introduce the concept of Bellman-consistent pessimism. It enables the sample-efficient guarantees with only the Bellman-completeness assumption which is standard in the exploratory setting, whereas the point-wise/bonus-based pessimism popularly adopted in the literature usually requires stronger and/or extra assumptions. Algorithmically, we demonstrate how to implicitly infer a policy value lower bound through a version space and provide a tractable implementation. A particularly important aspect of our results is the ability to adapt to the best bias-variance tradeoff in the hindsight, which no prior algorithms achieve to the best of our knowledge. When applying our results in linear MDPs, we attain an $\mathcal{O}(d)$ improvement in sample complexity, compared with the best-known recent work of offline RL in linear MDPs, whenever the action space is finite and small.

As of limitations and future work, the sample complexity of our practical algorithm is worse than that of the information-theoretic approach, and it will be interesting to close this gap. Another future direction is to empirically evaluate PSPI on benchmarks and compare it to existing approaches.

References

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

[2] András Antos, Csaba Szepesvári, and Rémi Munos. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1):89–129, 2008.

[3] Sergey Levine, Aviral Kumar, George Tucker, and Justin Fu. Offline reinforcement learning: Tutorial, review, and perspectives on open problems. *arXiv preprint arXiv:2005.01643*, 2020.

[4] Jinglin Chen and Nan Jiang. Information-theoretic considerations in batch reinforcement learning. In *International Conference on Machine Learning*, pages 1042–1051, 2019.

[5] Ruosong Wang, Dean P Foster, and Sham M Kakade. What are the statistical limits of offline rl with linear function approximation? *arXiv preprint arXiv:2010.11895*, 2020.

[6] Ruosong Wang, Yifan Wu, Ruslan Salakhutdinov, and Sham M Kakade. Instabilities of offline rl with pre-trained neural representation. *arXiv preprint arXiv:2103.04947*, 2021.

[7] Rémi Munos. Error bounds for approximate policy iteration. In *Proceedings of the Twentieth International Conference on International Conference on Machine Learning*, pages 560–567, 2003.

[8] Csaba Szepesvári and Rémi Munos. Finite time bounds for sampling based fitted value iteration. In *Proceedings of the 22nd international conference on Machine learning*, pages 880–887, 2005.

[9] Rémi Munos and Csaba Szepesvári. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9(5), 2008.

[10] Amir Massoud Farahmand, Rémi Munos, and Csaba Szepesvári. Error propagation for approximate policy and value iteration. In *Advances in Neural Information Processing Systems*, 2010.

[11] Tengyang Xie and Nan Jiang. Q* approximation schemes for batch reinforcement learning: A theoretical comparison. *arXiv preprint arXiv:2003.03924*, 2020.

[12] Rahul Kidambi, Aravind Rajeswaran, Praneeth Netrapalli, and Thorsten Joachims. Morel: Model-based offline reinforcement learning. *arXiv preprint arXiv:2005.05951*, 2020.

[13] Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *International Conference on Machine Learning*, pages 2052–2062, 2019.

[14] Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. *arXiv preprint arXiv:2006.04779*, 2020.

[15] Yao Liu, Adith Swaminathan, Alekh Agarwal, and Emma Brunskill. Provably good batch reinforcement learning without great exploration. *arXiv preprint arXiv:2007.08202*, 2020.

[16] Ying Jin, Zhuoran Yang, and Zhaoran Wang. Is pessimism provably efficient for offline rl? *arXiv preprint arXiv:2012.15085*, 2020.

[17] Eyal Even-Dar, Sham M Kakade, and Yishay Mansour. Online markov decision processes. *Mathematics of Operations Research*, 34(3):726–736, 2009.

[18] Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. In *International Conference on Machine Learning*, pages 2160–2169. PMLR, 2019.

[19] Alekh Agarwal, Sham M Kakade, Jason D Lee, and Gaurav Mahajan. On the theory of policy gradient methods: Optimality, approximation, and distribution shift. *arXiv preprint arXiv:1908.00261*, 2019.

[20] Dimitri P. Bertsekas and John N. Tsitsiklis. *Neuro-Dynamic Programming*. Athena Scientific, 1st edition, 1996. ISBN 1886529108.

[21] Vijay R Konda and John N Tsitsiklis. Actor-critic algorithms. In *Advances in neural information processing systems*, pages 1008–1014. Citeseer, 2000.

[22] Tengyang Xie and Nan Jiang. Batch value-function approximation with only realizability. *arXiv preprint arXiv:2008.04990*, 2020.

[23] Yaqi Duan, Zeyu Jia, and Mengdi Wang. Minimax-optimal off-policy evaluation with linear function approximation. In *International Conference on Machine Learning*, pages 2701–2709. PMLR, 2020.

[24] Aviral Kumar, Justin Fu, George Tucker, and Sergey Levine. Stabilizing off-policy q-learning via bootstrapping error reduction. *arXiv preprint arXiv:1906.00949*, 2019.

[25] Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn, and Tengyu Ma. Mopo: Model-based offline policy optimization. *arXiv preprint arXiv:2005.13239*, 2020.

[26] Bernardo Ávila Pires and Csaba Szepesvári. Statistical linear estimation with penalized estimators: an application to reinforcement learning. In *Proceedings of the 29th International Coference on International Conference on Machine Learning*, pages 1755–1762, 2012.

[27] Alessandro Lazaric, Mohammad Ghavamzadeh, and Rémi Munos. Finite-sample analysis of least-squares policy iteration. *Journal of Machine Learning Research*, 13:3041–3074, 2012.

[28] Sham Kakade and John Langford. Approximately optimal approximate reinforcement learning. In *ICML*, volume 2, pages 267–274, 2002.

[29] Bruno Scherrer. Approximate policy iteration schemes: a comparison. In *International Conference on Machine Learning*, pages 1314–1322. PMLR, 2014.

[30] Romain Laroche, Paul Trichelair, and Remi Tachet Des Combes. Safe policy improvement with baseline bootstrapping. In *International Conference on Machine Learning*, pages 3652–3661, 2019.

[31] Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pages 2137–2143. PMLR, 2020.

[32] Qi Cai, Zhuoran Yang, Chi Jin, and Zhaoran Wang. Provably efficient exploration in policy optimization. In *International Conference on Machine Learning*, pages 1283–1294. PMLR, 2020.

[33] Lior Shani, Yonathan Efroni, Aviv Rosenberg, and Shie Mannor. Optimistic policy optimization with bandit feedback. In *International Conference on Machine Learning*, pages 8604–8613. PMLR, 2020.

[34] Amit Daniely, Sivan Sabato, Shai Ben-David, and Shai Shalev-Shwartz. Multiclass learnability and the erm principle. In *Proceedings of the 24th Annual Conference on Learning Theory*, pages 207–232. JMLR Workshop and Conference Proceedings, 2011.

[35] Shai Bendavid, Nicolo Cesabianchi, David Haussler, and Philip M Long. Characterizations of learnability for classes of $\{0, \ldots, n\}$-valued functions. *Journal of Computer and System Sciences*, 50(1):74–86, 1995.

[36] David Haussler and Philip M Long. A generalization of sauer's lemma. *Journal of Combinatorial Theory, Series A*, 71(2):219–240, 1995.

[37] David Haussler. Sphere packing numbers for subsets of the boolean n-cube with bounded vapnik-chervonenkis dimension. *Journal of Combinatorial Theory, Series A*, 69(2):217–232, 1995.

[38] Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. Contextual decision processes with low bellman rank are pac-learnable. In *International Conference on Machine Learning*, pages 1704–1713. PMLR, 2017.

[39] Mehryar Mohri, Afshin Rostamizadeh, and Ameet Talwalkar. *Foundations of machine learning*. MIT press, 2018.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section 5.
   (c) Did you discuss any potential negative societal impacts of your work? [N/A]
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [Yes]
   (b) Did you include complete proofs of all theoretical results? [Yes]
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [N/A]
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [N/A]
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [N/A]
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [N/A]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...
   (a) Did you include the full text of instructions given to participants and screenshots, if applicable? [N/A]
   (b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]
   (c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]