Information-Theoretic Considerations in Batch Reinforcement Learning

Jinglin Chen [1] **Nan Jiang** [1]

Abstract

Value-function approximation methods that operate in batch mode have foundational importance to reinforcement learning (RL). Finite sample guarantees for these methods often crucially rely on two types of assumptions: (1) mild distribution shift, and (2) representation conditions that are stronger than realizability. However, the necessity ("why do we need them?") and the naturalness ("when do they hold?") of such assumptions have largely eluded the literature. In this paper, we revisit these assumptions and provide theoretical results towards answering the above questions, and make steps towards a deeper understanding of value-function approximation.

1. Introduction and Related Work

We are concerned with value-function approximation in batch-mode reinforcement learning, which is related to and sometimes known as Approximate Dynamic Programming (ADP; Bertsekas & Tsitsiklis, 1996). Such methods take sample transition data as input[1] and approximate the optimal value-function $Q^\star$ from a restricted class that encodes one's prior knowledge and inductive biases. They provide an important foundation for RL's empirical success today, as many popular deep RL algorithms find their prototypes in this literature. For example, when DQN (Mnih et al., 2015) is run on off-policy data, and the target network is updated slowly, it can be viewed as the stochastic approximation of its batch analog, Fitted Q-Iteration (Ernst et al., 2005), with a neural net as the function approximator (Riedmiller, 2005; Yang et al., 2019).

Given the importance of these methods, the question of *when they work* is central to our understanding of RL. Existing works that analyze error propagation and finite sample behavior of ADP methods (Munos, 2003; Szepesvári & Munos, 2005; Antos et al., 2008; Munos & Szepesvári, 2008; Tosatto et al., 2017) have provided us with a decent understanding: To guarantee sample-efficient learning of near-optimal policies, we often need assumptions from the following two categories.

Mild distribution shift Many ADP methods can run completely off-policy and they do the best with whatever data available.[2] Therefore, it is necessary that the data have sufficient coverage over the state (and action) space.

Representation condition Since the ultimate goal is to find $Q^\star$, we would expect that the function class we work with contains it (or at least a close approximation). While such realizability-type assumptions are sufficient for supervised learning, reinforcement learning faces the additional difficulties of delayed consequences and the lack of labels, and existing analyses often make stronger assumptions on the function class, such as (approximate) closedness under Bellman update (Szepesvári & Munos, 2005).

While the above assumptions make intuitive sense, and finite sample bounds have been proved when they hold, their necessity ("*can we prove similar results without making these assumptions?*") and naturalness ("*do they actually hold in interesting problems?*") have largely eluded the literature. In this paper, we revisit these assumptions and provide theoretical results towards answering the above questions. Below is a highlight of our results:

1. To prepare for later discussions, we provide an analysis of representative ADP algorithms (FQI and its variant) under a simplified and minimal setup (Section 3). As a side-product, our results improve upon prior analyses in the dependence of error rate on sample size.

2. We formally justify the necessity of mild distribution shift via an information-theoretic lower bound (Section 4.1). Our setup rules out trivial and uninteresting failure mode due to an adversarial choice of data: Even

[1] University of Illinois at Urbana-Champaign, Urbana, Illinois, USA. Correspondence to: Nan Jiang <nanjiang@illinois.edu>.

Proceedings of the 36^{th} International Conference on Machine Learning, Long Beach, California, PMLR 97, 2019. Copyright 2019 by the author(s).

[1] In this paper, we restrict ourselves to the so-called *one-path* setting and do not allow multiple samples from the same state (Sutton & Barto, 1998; Maillard et al., 2010), which is only feasible in certain simulated environments and allows algorithms to succeed with realizability as the only representation condition.

[2] Even when they are on-policy or combined with a standard exploration module (e.g., ϵ-greedy), most often they fail in problems where exploration is difficult (e.g., combination lock; see Kakade, 2003) and rely on the benignness of data to succeed.

with the most favorable data distribution, polynomial sample complexity is not achievable if the MDP dynamics are not restricted.

3. We conjecture an information-theoretic lower bound against realizability alone as the representation condition (Conjecture 8, Section 5.1). While we are not able to prove the conjecture, important steps are made, as two very general proof styles are shown to be destined to fail, one of which is due to Sutton & Barto (2018) and has been used to prove a closely related result.

4. As another side-product, we prove that *model-based RL* can enjoy polynomial sample complexity with realizability alone (Corollary 6). If Conjecture 8 is true, we have a formal separation showing the gap between batch model-based vs value-based RL with function approximation (see the analog in the online exploration setting in Sun et al. (2019)).

Throughout the paper, we make novel connections to two subareas of RL: state abstractions (Whitt, 1978; Li et al., 2006) and PAC exploration under function approximation (Krishnamurthy et al., 2016; Jiang et al., 2017). In particular, we are able to utilize some of their results in our proofs (Sections 4.1 and 5.1), and find examples from these areas where the assumptions of interest hold (Sections 4.2 and 5.2). This suggests that the results in these other areas may be beneficial to the research in ADP, and we hope this work can inspire researchers from different subareas of RL to exchange ideas more often.

2. Preliminaries

2.1. Markov Decision Processes (MDPs)

Let $M = (\mathcal{S}, \mathcal{A}, P, R, \gamma, \eta_1)$ be an MDP, where $\mathcal{S}$ is the finite (but can be arbitrarily large) state space, $\mathcal{A}$ is the finite action space, $P : \mathcal{S} \times \mathcal{A} \to \Delta(\mathcal{S})$ is the transition function ($\Delta(\cdot)$ is 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 η_1 is the initial distribution over states.

A (stochastic) policy $\pi : \mathcal{S} \to \Delta(\mathcal{A})$ prescribes a distribution over actions for each state. Fixing a start state s, the policy π induces a random trajectory $s_1, a_1, r_1, s_2, a_2, r_2, \ldots$, where $s_1 = s$, $a_1 \sim \pi(s_1)$, $r_1 = R(s_1, a_1)$, $s_2 \sim P(s_1, a_1)$, $a_2 \sim \pi(s_2)$, etc. The goal is to find π that maximizes the expected return $v^\pi := \mathbb{E}[\sum_{h=1}^\infty \gamma^{h-1} r_h | s_1 \sim \eta_1, \pi]$. It will also be useful to define the value function $V^\pi(s) := \mathbb{E}[\sum_{h=1}^\infty \gamma^{h-1} r_h | s_1 = s, \pi]$ and Q-value function $Q^\pi(s, a) := \mathbb{E}[\sum_{h=1}^\infty \gamma^{h-1} r_h | s_1 = s, a_1 = a, a_{2:\infty} \sim \pi]$, and these functions take values in $[0, V_{\max}]$ with $V_{\max} := R_{\max}/(1 - \gamma)$.

There exists a deterministic policy[3] $\pi^\star$ that maximizes $V^\pi(s)$ for all $s \in \mathcal{S}$ simultaneously, and hence also maximizes v^π as $v^\pi = \mathbb{E}_{s_1 \sim \eta_1}[V^\pi(s_1)]$. Let $V^\star$ and $Q^\star$ be the shorthand for $V^{\pi^\star}$ and $Q^{\pi^\star}$ respectively. It is well known that $\pi^\star(s) = \pi_{Q^\star}(s) := \arg\max_{a \in \mathcal{A}} Q^\star(s, a)$, and $Q^\star$ satisfies the *Bellman equation* $Q^\star = \mathcal{T} Q^\star$, where $\mathcal{T} : \mathbb{R}^{\mathcal{S} \times \mathcal{A}} \to \mathbb{R}^{\mathcal{S} \times \mathcal{A}}$ is the *Bellman update operator*: $\forall f \in \mathbb{R}^{\mathcal{S} \times \mathcal{A}}$,

$$(\mathcal{T} f)(s, a) := R(s, a) + \gamma \mathbb{E}_{s' \sim P(s,a)}[V_f(s')], \quad (1)$$

where $V_f(s') := \max_{a' \in \mathcal{A}} f(s', a')$.

Additional notations Let η_h^π be the marginal distribution of s_h under π, that is, $\eta_h^\pi(s) := \Pr[s_h = s \mid s_1 \sim \eta_1, \pi]$. For $g : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$, $\nu \in \Delta(\mathcal{S} \times \mathcal{A})$, and $p \geq 1$, define the shorthand $\|g\|_{p,\nu} := (\mathbb{E}_{(s,a)\sim\nu}[|g(s, a)|^p])^{1/p}$, which is a semi-norm. Furthermore, for any object that is a function of/distribution over $\mathcal{S}$ (or $\mathcal{S} \times \mathcal{A}$), we will treat it as a vector whenever convenient. We add a subscript to the value functions or Bellman update operators, e.g., $V_M^\star$, when it is necessary to clarify the MDP in which the object is defined.

2.2. Batch Value-Function Approximation

This paper is concerned with *batch-mode* RL with value-function approximation. As a typical setup, the agent does not have direct access to the MDP and instead is given the following inputs:

- A batch dataset D consisting of (s, a, r, s') tuples, where $r = R(s, a)$ and $s' \sim P(s, a)$. For simplicity, we assume that (s, a) is generated i.i.d. from the *data distribution* $\mu \in \Delta(\mathcal{S} \times \mathcal{A})$.[4]
- A class of candidate value-functions, $\mathcal{F} \subset (\mathcal{S} \times \mathcal{A} \to [0, V_{\max}])$, which (approximately) captures $Q^\star$; such a property is often called *realizability*. We discuss additional assumptions on $\mathcal{F}$ later. As a further simplification, we focus on finite but exponentially large $\mathcal{F}$ and discuss how to handle infinite classes when appropriate.

The learning goal is to compute a near-optimal policy from the data, often via finding $f \in \mathcal{F}$ that approximates $Q^\star$ and outputting π_f, the greedy policy w.r.t. f. A representative algorithm for this setting is Fitted Q-Iteration (FQI) (Ernst et al., 2005; Szepesvári, 2010).[5] The algorithm initializes

[3] A deterministic policy puts all the probability mass on a single action in each state. With a slight abuse of notation, we sometimes also treat the type of such policies as $\pi : \mathcal{S} \to \mathcal{A}$.

[4] The agent may or may not have knowledge of μ. Most existing algorithms are agnostic to such knowledge.

[5] Batch value-based algorithms can often be categorized into approximate value iteration (e.g., FQI) and approximate policy iteration (e.g., LSPI (Lagoudakis & Parr, 2003)). We focus on the former due to its simplicity and do not discuss the latter as its

$f_0 \in \mathcal{F}$ arbitrarily, and iteratively computes f_k as follows: in iteration k, the algorithm converts the dataset D into a regression dataset, with (s,a) being the input and $r + \gamma V_{f_{k-1}}(s')$ as the output. It then minimizes the squared loss regression objective over $\mathcal{F}$, and the minimizer becomes f_k. More formally, $f_k := \widehat{\mathcal{T}}_{\mathcal{F}} f_{k-1}$, where

$$\widehat{\mathcal{T}}_{\mathcal{F}} f' := \arg\min_{f \in \mathcal{F}} \mathcal{L}_D(f; f') \tag{2}$$

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

FQI may oscillate and a fixed point solution may not exist in general (Gordon, 1995). Nevertheless, under conditions which we will specify later, finite sample guarantees for FQI can still be obtained even if the process does not converge.

2.3. State Abstractions

A state abstraction ϕ maps $\mathcal{S}$ to a finite and potentially much smaller abstract state space, $\mathcal{S}_\phi$. Naturally, ϕ is often a many-to-one mapping, inducing an equivalence notion over $\mathcal{S}$ which encodes one's prior knowledge of equivalent or similar states. A typical use of abstractions in the batch learning setting is to construct a tabular (or *certainty-equivalent*) model from a dataset $\{(\phi(s), a, r, \phi(s'))\}$, and compute the optimal policy in the resulting abstract model. There is a long history of studying abstractions, mostly focusing on their approximation guarantees (Whitt, 1978).

We note, however, that there is a direct connection between FQI and certainty-equivalence with abstractions. In particular, value iteration in the model estimated with abstraction ϕ is *exactly equivalent* to FQI with $\mathcal{F}$ being the class of piece-wise constant functions under ϕ.[6] As such, the characterization of approximation errors in the two bodies of literature are closely related to each other. We will discuss further connections in the rest of this paper.

3. Bellman Error Minimization in Batch Reinforcement Learning

In this section, we give a complete analysis of FQI and a related algorithm, with the main results being two sample complexity bounds. Many of the insights and results in this section have either explicitly appeared in or been implicitly hinted by prior work (especially Szepesvári & Munos, 2005; Antos et al., 2008), and we include them because (1) the discussions in the rest of the paper are largely based on these results, and (2) our analyses simplify prior results without trivializing them, making the high-level insights more accessible. We also improve the results in some aspects.

3.1. Sample-Based Bellman Error Minimization

We start by deriving FQI from a slightly unusual perspective due to the aforementioned prior work, which motivates major assumptions in FQI analysis and introduces concepts that are important for later discussions.

Recall that the goal of value-based RL is to find $f \in \mathcal{F}$ such that $f \approx \mathcal{T} f$, that is, $\|f - \mathcal{T} f\| = 0$ where $\|\cdot\|$ is some appropriate norm. For example, if μ is a distribution supported on the entire $\mathcal{S} \times \mathcal{A}$, then $\|f - \mathcal{T} f\|_{2,\mu}^2 = 0$ would guarantee that $f = Q^\star$. While such an f can be found in principle by minimizing $\|f - \mathcal{T} f\|_{2,\mu}^2$ over $f \in \mathcal{F}$, calculating $\|f - \mathcal{T} f\|$ requires knowledge of the transition dynamics (recall Eq.(1)), which is unknown in the learning setting. Instead, we have access to the dataset $D = \{(s,a,r,s')\}$, and it may be tempting to minimize the following objective that is purely a function of data: (Recall $\mathcal{L}_D$ in Eq.(2))

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

Unfortunately, even with the infinite amount of data, the above objective is still different from the actual Bellman error $\|f - \mathcal{T} f\|_{2,\mu}^2$ that we wish to minimize. In particular, define $\mathcal{L}_\mu(\cdot;\cdot) := \mathbb{E}[\mathcal{L}_D(\cdot;\cdot)]$, where the expectation is w.r.t. the random draw of the dataset D. We have $\mathcal{L}_\mu(f; f) =$

$$\|f - \mathcal{T} f\|_{2,\mu}^2 + \gamma^2 \mathbb{E}_{(s,a)\sim\mu}[\mathbb{V}_{s' \sim P(s,a)}[V_f(s')]]. \tag{3}$$

In words, $\mathcal{L}_\mu(f; f)$ adds a conditional variance term to the desired objective, which incorrectly penalizes functions that have a large variance w.r.t. random state transitions.

The minimax algorithm [7] One way to fix the issue is to estimate the conditional variance term in Eq. (3) and subtracting it from $\mathcal{L}_D(f; f)$. In fact, it is easy to verify that $\gamma^2 \mathbb{E}_{(s,a)\sim\mu}[\mathbb{V}_{s'\sim P(s,a)}[V_f(s')]]$ is the Bayes optimal error of the regression problem

$$(s,a) \mapsto r + \gamma V_f(s'). \tag{4}$$

One can estimate it by empirical risk minimization over a rich function class, and the estimate is consistent as long as the function class realizes the Bayes optimal regressor and has bounded statistical complexity. Following this idea, we assume access to another function class $\mathcal{G} \subset (\mathcal{S} \times \mathcal{A} \to [0, V_{\max}])$ for solving the regression problem in Eq.(4). The estimated Bayes optimal error is

$$\inf_{g \in \mathcal{G}} \mathcal{L}_D(g; f). \tag{5}$$

guarantees often rely on similar but more complicated assumptions (Lazaric et al., 2012). Moreover, our lower bounds are information-theoretic and algorithm-independent.

[6] This result is known anecdotally (see e.g., Pires & Szepesvári, 2016) and we include details in Appendix E for completeness.

[7] Also known under the name "modified Bellman Residual Minimization" (Antos et al., 2008).

A good approximation to $\|f - \mathcal{T}f\|_{2,\mu}^2$ from data is then $\sup_{g\in\mathcal{G}} \mathcal{L}_D(f;f) - \mathcal{L}_D(g;f)$. This suggests that we can simply run the following optimization problem to find $f \in \mathcal{F}$ that approximates $Q^\star$:

$$\inf_{f\in\mathcal{F}} \sup_{g\in\mathcal{G}} \mathcal{L}_D(f;f) - \mathcal{L}_D(g;f). \tag{6}$$

Later in this section, we will provide a finite sample analysis of the above minimax algorithm, but before that, we will show that FQI can be viewed as its approximation.

FQI as an approximation to Eq.(6) FQI has a close connection to the above program and can be viewed as its approximation, when $\mathcal{G}$ is chosen to be $\mathcal{F}$. Formally,

Proposition 1. *Let $\hat{f}$, $\hat{g}$ be the solution to Eq.(6) when $\mathcal{G} = \mathcal{F}$.*

- *If $\mathcal{L}_D(\hat{f};\hat{f}) - \mathcal{L}_D(\hat{g};\hat{f}) = 0$, $\hat{f}$ is a fixed point for FQI.*
- *Conversely, if $f_k = f_{k-1}$ holds for some k in FQI, then $\hat{f} = \hat{g} = f_k$ is a solution to Eq.(6).*
- *If $\mathcal{L}_D(\hat{f};\hat{f}) - \mathcal{L}_D(\hat{g};\hat{f}) > 0$, FQI oscillates and no fixed point exists.*

The proof is deferred to Appendix A. The proposition states that the minimax algorithm is more stable than FQI, and when FQI reaches a fixed point, the solutions of the two algorithms coincide. In fact, Dai et al. (2018) derives a closely related algorithm using Fenchel dual and shows that the algorithm is always convergent.

3.2. Analysis of FQI and Its Minimax Variant

We provide finite sample guarantees to the two algorithms introduced above; closely related analyses have appeared in prior works (see Section 1 for references), and our version provides a cleaner analysis under simplification assumptions, improves the error rate as a function of sample size, and prepares us for later discussions.

To state the guarantees, we need to introduce the two assumptions that are core to this paper. The first assumption handles distribution shift, and we precede it with the definition of *admissible distributions*.

Definition 1 (Admissible distributions)**.** *We say a distribution $\nu \in \Delta(\mathcal{S}\times\mathcal{A})$ is* admissible *in MDP $M = (\mathcal{S}, \mathcal{A}, P, R, \gamma, \eta_1)$, if there exists $h \geq 0$, and a (potentially non-stationary and stochastic) policy π, such that $\nu(s,a) = \Pr[s_h = s, a_h = a | s_1 \sim \eta_1, \pi]$.*

Intuitively, a distribution is admissible if it can be generated in the MDP by following some policy for a number of timesteps. The following assumption on *concentratability* asserts that all admissible distributions are not "far away" from the data distribution μ. The original definition is due to Munos (2003).

Assumption 1 (Concentratability coefficient)**.** *We assume that there exists $C < \infty$ s.t. for any admissible ν,*

$$\forall (s,a) \in \mathcal{S}\times\mathcal{A},\ \frac{\nu(s,a)}{\mu(s,a)} \leq C.$$

The real (and implicit) assumption here is that C is manageably large, as our sample complexity bounds scale linearly with C. Prior works have used more sophisticated definitions (Farahmand et al., 2010).[8] The technicalities introduced are largely orthogonal to the discussions in this paper, so we choose to adopt a much simplified version. Despite the simplification, we will see natural examples that yield small C under our definition in Section 4. We will also discuss how to relax it using the structure of $\mathcal{F}$ at the end of the paper.

Next, we introduce the assumption on the representation power of $\mathcal{F}$ and $\mathcal{G}$.

Assumption 2 (Realizability)**.** $Q^\star \in \mathcal{F}$.
(When this holds approximately, we measure violation by $\epsilon_\mathcal{F} := \inf_{f\in\mathcal{F}} \|f - \mathcal{T}f\|_{2,\mu}^2$.)

Assumption 3 (Completeness)**.** $\forall f \in \mathcal{F}, \mathcal{T}f \in \mathcal{G}$.
(When this holds approximately, we measure violation by $\epsilon_{\mathcal{F},\mathcal{G}} := \sup_{f\in\mathcal{F}} \inf_{g\in\mathcal{G}} \|g - \mathcal{T}f\|_{2,\mu}^2$.)

These assumptions lead to finite sample guarantees for both the minimax algorithm and FQI. For FQI, since $\mathcal{G} = \mathcal{F}$, Assumption 3 essentially states that $\mathcal{F}$ *is closed under operator $\mathcal{T}$*, hence "completeness".[9] The assumption is natural from how we derive the minimax algorithm in Sec 3.1, as Eq.(5) is only a consistent estimate of the Bayes optimal error of Eq.(4) if $\mathcal{G}$ realizes the Bayes optimal regressor, which is $\mathcal{T}f$.

A few remarks in order:

1. When $\mathcal{F} = \mathcal{G}$ is finite, completeness implies realizability.[10] However, completeness is stronger and much less desired than realizability: realizability is monotone in $\mathcal{F}$ (adding functions to $\mathcal{F}$ never hurts realizability), while completeness is not (adding functions to $\mathcal{F}$ may break completeness).

2. While we focus on completeness, it is not the only condition that leads to guarantees for ADP algorithms. We discuss alternative assumptions in Section 6.

[8] This often comes at the cost of their bound being not *a priori*, i.e., having a dependence on the randomness of data, initialization, and tie-breaking in optimization.

[9] In the literature, the violation of completeness when $\mathcal{F} = \mathcal{G}$, $\epsilon_{\mathcal{F},\mathcal{F}}$, is called *inherent Bellman error*.

[10] This is because $\mathcal{T}^k f$ never repeats itself, as its ℓ_∞ distance to $Q^\star$ shrinks exponentially with a rate of γ due to contraction.

Now we are ready to state the sample complexity results. In Appendices C and D we provide more general error bounds (Theorems 11 and 17) that handle the approximate case where $\epsilon_{\mathcal{F}}$ and $\epsilon_{\mathcal{F},\mathcal{G}}$ are not zero and iteration k is finite. To keep the main text focused and accessible, we only present their sample complexity corollaries in the exact case.

Theorem 2 (Sample complexity of FQI)**.** *Given a dataset $D = \{(s, a, r, s')\}$ with sample size $|D| = n$ and $\mathcal{F}$ that satisfies completeness (Assumption 3 when $\mathcal{G} = \mathcal{F}$), w.p. $\geq 1 - \delta$, the output policy of FQI after k iterations, π_{f_k}, satisfies $v^\star - v^{\pi_{f_k}} \leq \epsilon \cdot V_{\max}$ when $k \to \infty$ and*[11] $n = O\left(\frac{C \ln \frac{|\mathcal{F}|}{\delta}}{\epsilon^2 (1-\gamma)^4}\right)$.

Theorem 3 (Sample complexity of the minimax variant)**.** *Given a dataset $D = \{(s, a, r, s')\}$ with sample size $|D| = n$ and $\mathcal{F}$, $\mathcal{G}$ that satisfy realizability (Assumption 2) and completeness (Assumption 3) respectively, w.p. $\geq 1 - \delta$, the output policy of the minimax algorithm (Eq.(6)), $\pi_{\hat{f}}$, satisfies $v^\star - v^{\pi_{\hat{f}}} \leq \epsilon \cdot V_{\max}$, if $n = O\left(\frac{C \ln \frac{|\mathcal{F}||\mathcal{G}|}{\delta}}{\epsilon^2 (1-\gamma)^4}\right)$.*

Our results show that the suboptimality ϵ decreases in the rate of $n^{-1/2}$ when realizability and completeness hold exactly, and the more general error bounds (Theorems 11 and 17) degrade gracefully from the exact case as $\epsilon_{\mathcal{F},\mathcal{F}}$ (or $\epsilon_{\mathcal{F}}$ and $\epsilon_{\mathcal{F},\mathcal{G}}$) increases. This is obtained via the use of Bernstein's inequality to achieve fast rate in least square regression. While results similar to Theorems 2 and 11 exist (Farahmand 2011, Chapter 5; see also Lazaric et al. (2012); Pires & Szepesvári (2012); Farahmand et al. (2016)), according to our knowledge, fast rate for the minimax algorithm has not been established before: for example, Antos et al. (2008); Munos & Szepesvári (2008) obtain an error rate of $n^{-1/4}$ in closely related settings, but their rates do not improve to $n^{-1/2}$ in the absence of approximation.[12] The major limitation of our result is the assumption of finite $\mathcal{F}$ and $\mathcal{G}$ due to our minimal setup, and we refer readers to Yang et al. (2019) for a recent analysis that specializes in ReLU networks.[13]

We do not discuss the proofs in further details since the improvement in error rate is a side-product and this section is mainly meant to simplify prior analyses and provide a basis for subsequent discussions. Interested readers are invited to consult Appendices C and D where we provide sketched outlines as well as detailed proofs.

4. On Concentratability

In this section, we establish the necessity of Assumption 1 and show natural examples where concentratability is low. While it is easy to construct a counterexample of missing data[14] against removing Assumption 1, such a counterexample only reflects a trivial failure mode due to an adversarial choice of data. What we show is a deeper and nontrivial failure mode: Even with the *most favorable* data distribution, polynomial sample complexity is precluded if we put no restriction on MDP dynamics. This result improves our understanding on concentratability, and shows that this assumption is not only about the data distribution, but also (and perhaps more) about the environment and the state distributions induced therein.

4.1. Lower Bound

To show that low concentratability is necessary, we prove a hardness result, where both realizability and completeness hold, and an algorithm has the freedom to choose *any* data distribution μ that is favorable, yet *no algorithm* can achieve $poly(|\mathcal{A}|, \frac{1}{1-\gamma}, \ln|\mathcal{F}|, \ln|\mathcal{G}|, \frac{1}{\epsilon}, \frac{1}{\delta})$ sample complexity. Crucially, the concentratability coefficient of any data distribution on the worst-case MDP is always exponential in horizon, so the lower bound does not conflict with the upper bounds in Section 3, as the exponential sample complexity would have been explained away by the dependence on C.

Theorem 4. *There exists a family of MDPs $\mathcal{M}$ (they share the same $\mathcal{S}$, $\mathcal{A}$, γ), $\mathcal{F}$ that realizes the $Q^\star$ of every MDP in the family, and $\mathcal{G}$ that realizes $\mathcal{T}_{M'} f$ for any $M' \in \mathcal{M}$ and any $f \in \mathcal{F}$, such that: for any data distribution and any batch algorithm with $(\mathcal{F}, \mathcal{G})$ as input, an adversary can choose an MDP from the family, such that the sample complexity for the algorithm to find an ϵ-optimal policy cannot be poly$(|\mathcal{A}|, \frac{1}{1-\gamma}, \ln|\mathcal{F}|, \ln|\mathcal{G}|, \frac{1}{\epsilon}, \frac{1}{\delta})$.*

Proof. We construct $\mathcal{M}$, a family of hard MDPs, and prove the theorem via the combination of two arguments:

1. All algorithms are subject to an exponential lower bound (w.r.t. the horizon) even if (a) they have compact $\mathcal{F}$ and $\mathcal{G}$ that satisfy realizability and completeness as inputs, and (b) they can perform *exploration* during data collection.

2. Since the MDPs in the construction share the same deterministic transition dynamics, the combination of any data distribution and any batch RL algorithm is *a special case* of an exploration algorithm.

We first provide argument (1), which reuses the construction by Krishnamurthy et al. (2016). Let each instance of $\mathcal{M}$ be a complete tree with branching factor $|\mathcal{A}|$ and depth

[11] Only absolute constants are suppressed in Big-Oh notations.

[12] Note however that they handle infinite function classes. In fact, Munos & Szepesvári (2008, pg.831) have discussed the possibility of an $n^{-1/2}$ result, which we obtain here. See the beginning of Appendix C for further discussions.

[13] Their analysis modifies the FQI algorithm and samples fresh data in each iteration, dodging some of the technical difficulties due to reusing the same batch of data, which we handle here.

[14] That is, μ puts 0 probability on important states and actions.

$H = \lfloor 1/(1-\gamma) \rfloor$. Transitions are deterministic, and only leaf nodes have non-zero rewards. All leaves give $\text{Ber}(1/2)$ rewards, except for one that gives $\text{Ber}(1/2+\epsilon)$. Changing the position of this optimal leaf yields a family of $|\mathcal{A}|^H$ MDPs, and in order to achieve a suboptimality that is a constant fraction of ϵ, the algorithm is required to identify this optimal leaf.[15] In fact, the problem is equivalent to the hard instances of best arm identification with $|\mathcal{A}|^H$ arms, so even if an algorithm can perform active exploration, the sample complexity is still $\Omega(|\mathcal{A}|^H \ln(1/\delta)/\epsilon^2)$ (see Krishnamurthy et al. (2016) for details, who use standard techniques from Auer et al. (2002)).

Now we provide $\mathcal{F}$ and $\mathcal{G}$ that (1) satisfy Assumptions 2 and 3, (2) do not provide any information other than the fact that the problem is in $\mathcal{M}$, and (3) have "small" logarithmic sizes so that $\ln|\mathcal{F}|$ and $\ln|\mathcal{G}|$ cannot explain away the exponential sample complexity. Let $\mathcal{F} = \{Q^\star_{M'} : M' \in \mathcal{M}\}$, where the subscript specifies the MDP with respect to which we compute $Q^\star$. Let $\mathcal{G} = \{\mathcal{T}_{M'} Q^\star_{M''} : M', M'' \in \mathcal{M}\}$. Such $\mathcal{F}$ and $\mathcal{G}$ satisfy realizability and completeness by definition, and have statistical complexities $\ln|\mathcal{F}| = H \ln|\mathcal{A}|$ and $\ln|\mathcal{G}| \le 2H\ln|\mathcal{A}|$, respectively. With this, we conclude that any *exploration* algorithm cannot obtain $poly(|\mathcal{A}|, \frac{1}{1-\gamma}, \ln|\mathcal{F}|, \ln|\mathcal{G}|, \frac{1}{\epsilon})$ sample complexity.

We complete the proof with the second argument. Note that all the MDPs in $\mathcal{M}$ only differ in leaf rewards and share the same deterministic transition dynamics. Therefore, a learner with the ability to actively explore can mimic the combination of *any data distribution $\mu \in \Delta(\mathcal{S}\times\mathcal{A})$ and any batch RL algorithm*, by (1) collecting data from μ (which is always doable due to known and deterministic transitions), and (2) running the batch algorithm after data is collected. This completes the proof. □

4.2. Natural Examples

We have shown that polynomial learning is precluded if no restriction is put on the MDP dynamics, even if data is chosen in a favorable manner. The next question is, is low concentratability common, or at least found in interesting problems? In general, even if the data distribution μ is uniform over the state-action space, the worst-case C might still scale with $|\mathcal{S} \times \mathcal{A}|$, which can be too large in challenging RL problems for the guarantees to be any meaningful. To this end, Munos (2007) has provided several carefully constructed tabular examples, demonstrating that C does not always scale badly. However, are there more general problem families that capture RL scenarios found in empirical work, yet always yield a bounded C?

Example in problems with rich observations We find answers to the above problem in recent development of PAC exploration in rich-observation problems (Krishnamurthy et al., 2016; Jiang et al., 2017; Dann et al., 2018), where a general low-rank condition (a.k.a. *Bellman rank* (Jiang et al., 2017)) has been identified that enables sample-efficient exploration under function approximation. One of the prominent examples where such a condition holds is inspired by "visual gridworld" environments in empirical RL research (see e.g., Johnson et al., 2016): the dynamics are defined over a small number of hidden states (e.g., grids), and the agent receives high dimensional observations that are generated i.i.d. from the hidden states (e.g., raw-pixel images as observations). Below we show that in these environments, there always exists a data distribution that yields small C for batch learning, and such a distribution can be naturally generated as a mixture of admissible distributions. We include an informal statement below, deferring the precise version and the proof to Appendix B.

Proposition 5 (Informal)**.** *Let M be a reactive POMDP as defined in Jiang et al. (2017), where the underlying hidden state space $\mathcal{Z}$ is finite but the (Markov) observation space $\mathcal{S}$ can be arbitrarily large. There always exists a state-action distribution μ such that $C = |\mathcal{Z}\times\mathcal{A}|$ satisfies Assumption 1. Furthermore, μ can be obtained by taking a probability mixture of several admissible distributions.*

Similar results can be established for other structures studied by Jiang et al. (2017) (e.g., large MDPs with low-rank transitions), which we omit here. These results suggest that *Bellman rank is the counterpart for concentratability coefficient in the online exploration setting*. Further implications and how to leverage this connection to improve the definition of concentratability will be discussed in Section 6.

5. On Completeness

5.1. Towards an Information-Theoretic Lower Bound in the Absence of Completeness

We would also like to establish the necessity of completeness by showing that, there exist hard MDPs that cannot be efficiently learned with value-function approximation, even under low concentratability and realizability (Assumptions 1 and 2).[16] In fact, *algorithm-specific* hardness results have been known for a long time (see e.g., Van Roy, 1994; Gordon, 1995; Tsitsiklis & Van Roy, 1997), where ADP algorithms are shown to diverge even in MDPs with a small number of states, when the algorithm is forced to work with a restricted class of functions.[17] Unfortunately, such

[15] All leaf rewards are discounted by only a constant when $\gamma \to 1$, as $\gamma^{1/(1-\gamma)} \to e^{-1}$.

[16] Note that the existence of such a lower bound would not imply that completeness is indispensable. Rather it simply states that realizability alone is insufficient, and we need stronger conditions on $\mathcal{F}$, for which completeness is a candidate.

[17] Interested readers can consult Agrawal (2018). See also Dann et al. (2018, Theorem 45) for a more plain example.

hardness results are insufficient to confirm the fundamental difficulty of the problem, and it is important to seek *information-theoretic* lower bounds.

While we are not able to obtain such a lower bound, what we find is that the counterexample (if it exists) must be highly nontrivial and probably need ideas that are not present in standard statistical learning theory (SLT) and RL literature. More concretely, we show that two general proof styles are destined to fail in such a task, as polynomial sample complexity can be achieved information-theoretically.

Exponential-sized model family will not work Standard lower bounds in SLT often start with the construction of a family of problem instances that has an exponential size (Yu, 1997).[18] We show that this will simply never work, which is a direct corollary of Theorem 3:

Corollary 6 (Batch model-based RL only needs realizability)**.** *Let $D = \{(s, a, r, s')\}$ be a dataset with sample size $|D| = n$, C as defined in Assumption 1, and $\mathcal{M}$ a model class that realizes the true MDP M, i.e., $M \in \mathcal{M}$. There exists an (information-theoretic) algorithm that takes $\mathcal{M}$ as input and return an $(\epsilon V_{\max})$-optimal policy w.p. $\geq 1 - \delta$, if*

$$n = O\left(\frac{C \ln \frac{|\mathcal{M}|}{\delta}}{\epsilon^2 (1-\gamma)^4}\right).$$

Proof. We use the same idea as the proof of Theorem 4: Let $\mathcal{F} = \{Q^\star_{M'} : M' \in \mathcal{M}\}$, and $\mathcal{G} = \{\mathcal{T}_{M'} Q^\star_{M''} : M', M'' \in \mathcal{M}\}$. Note that $\ln |\mathcal{F}| \leq \ln |\mathcal{M}|$, and $\ln |\mathcal{G}| \leq 2 \ln |\mathcal{M}|$. $(\mathcal{F}, \mathcal{G})$ satisfy both realizability and completeness, so we apply the minimax algorithm (Eq.(6)) and the guarantee in Theorem 3 immediately holds. □

Essentially, this result shows that batch model-based RL can succeed with realizability as the only representation condition for the model class, because we can reduce it to value-based learning and obtain completeness *for free*. This illustrates a significant barrier to an algorithm-independent lower bound, that in an information-theoretic setting, the learner can always specialize in the family of hard instances and have the freedom to choose its algorithm style, thus can be *model-based*. However, in the context of value-function approximation, it is obvious that we are assuming no prior knowledge of the model class and hence cannot run any model-based algorithm. How can we encode such a constraint mathematically?

Tabular MDPs with a restricted value-function class will not work Sutton & Barto (2018, Section 11.6) proposes a clever way to prevent the learner to be model-based for linear function approximation, and a closely related definition is recently given by Sun et al. (2019) that applies to arbitrary function classes.

The idea is the following: Instead of providing the dataset $D = \{(s, a, r, s')\}$ directly, we preprocess the data and mask the identity of s (and s'). While s is not directly observable, the learner can query the evaluation of any $f \in \mathcal{F}$ on s for any $a \in \mathcal{A}$. That is, we represent each state s by its *value profile*, $\{f(s, a) : f \in \mathcal{F}, a \in \mathcal{A}\}$. This definition agrees with intuition and can be used to express a wide range of popular algorithms, including FQI.

Using this definition, Sutton & Barto (2018) proves a result closely related to what we aim at here: they show that the Bellman error $\|f - \mathcal{T} f\|$ is not learnable. In particular, there exist two MDPs (with finite and constant-sized state space) and a value function, such that (1) a value-based learner (who only has access to the value profiles of states) cannot distinguish between the data coming from the two MDPs, and (2) the Bellman error of the value function is different in the two MDPs.

While encouraging and promising, their constructions have a crucial caveat for our purpose, that the value function class is not realizable.[19] With further investigation, we sadly find that such a caveat is fundamental: no information-theoretic lower bound can be shown if realizability holds in naïve tabular constructions with a constant-sized state-action space and uniform data, hence value profile cannot be the *only* mechanism to induce hardness. In fact, we can prove a stronger result than we need here for $\mathcal{S}$ and $\mathcal{A}$ that are not necessarily constant-sized:

Proposition 7. *Let M be an MDP with a finite state space and $\mathcal{F}$ a realizable function class. Given a dataset $D = \{(s, a, r, s')\}$ where each (s, a) receives $\Omega(|D|/|\mathcal{S} \times \mathcal{A}|)$ samples, there exists an algorithm that only operates on states via their value profiles yet enjoy poly$(|\mathcal{S}|, |\mathcal{A}|, \frac{1}{1-\gamma}, \frac{1}{\epsilon}, \frac{1}{\delta})$ sample complexity.*

Proof Sketch. (See full proof in Appendix F.) If every $s \in \mathcal{S}$ has a unique value profile, the state is perfectly decodable and thus one can simply compute the optimal policy of the certainty-equivalent model. If a set of states share exactly the same value profile—and w.l.o.g. let's consider 2 states, s_1 and s_2—realizability implies that $Q^\star(s_1, a) = Q^\star(s_2, a)$, $\forall a \in \mathcal{A}$. Now consider the algorithm that treat all states with the same value profile as the same state, which essentially uses a state abstraction that is *$Q^\star$-irrelevant* (Li et al., 2006). It is known that certainty-equivalence with $Q^\star$-irrelevant abstraction is consistent and enjoys polynomial sample complexity when each state-action pair receives enough data (Li, 2009; Hutter, 2014; Jiang et al., 2015; Abel et al., 2016; Jiang, 2018). □

[18] In fact, our Theorem 4 also follows this style, whose construction is due to Krishnamurthy et al. (2016); Jiang et al. (2017).

[19] They force two states who have different optimal values to share the same features for linear function approximation.

Given that we fail to obtain the lower bound, a conjecture is made below and we hope to resolve it in future work.

Conjecture 8. *There exists a family of MDPs $\mathcal{M}$ that share the same $\mathcal{S}$, $\mathcal{A}$, and γ, such that: any algorithm with $\mathcal{F} = \{Q^\star_{M'} : M' \in \mathcal{M}\}$ as input that can only access states via value profiles cannot have poly($\frac{1}{1-\gamma}, C, \ln|\mathcal{F}|, \frac{1}{\epsilon}, \frac{1}{\delta}$) sample complexity.*

5.2. Connection to Bisimulation

As the last piece of technical result of this paper, we show that when $\mathcal{F}$ is a space of piece-wise constant functions under a partition induced by state abstraction ϕ, the notion of completeness (Assumption 3, $\mathcal{F} = \mathcal{G}$) is exactly equivalent to a long-studied type of abstractions, known as *bisimulation* (Whitt, 1978; Even-Dar & Mansour, 2003; Ravindran, 2004; Li et al., 2006).

Definition 2 (Bisimulation)**.** *An abstraction $\phi : \mathcal{S} \to \mathcal{S}_\phi$ is a bisimulation in an MDP M, if $\forall s_1, s_2$ where $\phi(s_1) = \phi(s_2)$ (i.e., they are aggregated), $R(s_1, a) = R(s_2, a)$ and $\sum_{s \in \phi^{-1}(x)} P(s|s_1, a) = \sum_{s \in \phi^{-1}(x)} P(s|s_2, a)$ for all $a \in \mathcal{A}$, $x \in \mathcal{S}_\phi$.*

Definition 3 (Piece-wise constant function class)**.** *Given an abstraction ϕ, define $\mathcal{F}^\phi \subset (\mathcal{S} \times \mathcal{A} \to [0, V_{\max}])$ as the set of all functions f that are piece-wise constant under ϕ. That is, $\forall s_1, s_2 \in \mathcal{S}$ where $\phi(s_1) = \phi(s_2)$, we have $f(s_1, a) = f(s_2, a), \forall a \in \mathcal{A}$.*

Proposition 9. *ϕ is bisimulation $\Leftrightarrow$ $\mathcal{F}^\phi$ satisfies completeness (Assumption 3 with $\mathcal{F} = \mathcal{G} = \mathcal{F}^\phi$).*

The "$\Rightarrow$" part is trivial, but the "$\Leftarrow$" part is less obvious. The proof shows that if ϕ is not a bisimulation, we can find $f \in \mathcal{F}^\phi$ either to witness the reward error or the transition error, and in the latter case, the choice of f achieves the maximum discrepancy in an integral probability metric (Müller, 1997) interpretation of the bisimulation condition on transition dynamics. Details are provided in Appendix E, where we prove a stronger result that relates the approximation error of bisimulation to the violation of completeness.

6. Discussions and Related Work

In this paper, we examine the common assumptions that enable finite sample guarantees for value-function approximation methods. Concretely, we provide an information-theoretic lower bound in Section 4.1, showing that not constraining the concentratability coefficient C immediately precludes sample-efficient learning even with benign data. We also introduce a general family of problems of interest in empirical RL that yield low concentratability (Section 4.2).

In comparison, the necessity of completeness is still a mystery, and our investigation in Section 5.1 mostly shows the highly nontrivial nature of the lower bound (assuming it exists) as we eliminate two general proof styles. We hope these negative results can guide the search for novel constructions that reflect the fundamental difficulties of reinforcement learning in the function approximation setting.

We conclude the paper with some discussions.

Alternative assumptions to completeness As we note in Section 5.1, even if Conjecture 8 is true, it would not imply that completeness is absolutely necessary, as other assumptions may also break the lower bound. Furthermore, additional assumptions are not necessarily made on the value-function class (e.g., that $\widehat{\mathcal{T}}_{\mathcal{F}}$ being a contraction (Gordon, 1995; Szepesvári & Smart, 2004; Lizotte, 2011; Pires & Szepesvári, 2016)), and can instead take the form of requiring another function class to realize other objects of interest, such as state distributions (Chen et al., 2018; Liu et al., 2018). Regardless, all of these approaches face the same fundamental question on the necessity of the additional/stronger assumptions being made, to which our Conjecture 8 is an important piece if not the final answer. We hope to resolve this important open question in the future.

Related work that has not been covered The conjectured insufficiency of realizability (Conjecture 8) is related to various undesirable phenomena in learning with bootstrapped targets, which has been of constant interest to RL researchers (Sutton, 2015; Van Hasselt et al., 2018; Lu et al., 2018). As far as we know, all existing efforts that investigate this issue are algorithm-specific (apart from Sutton & Barto (2018, Section 11.6) and the references therein, which has been discussed in Section 5.1), and our information-theoretic perspective is novel.

Relaxation of Assumption 1 using the structure of $\mathcal{F}$ The concentratability coefficient C is defined as a function of the MDP, even in its most complicated version (Farahmand et al., 2010). In Section 4.2 we discover a connection to *Bellman rank* (Jiang et al., 2017), which can be viewed as its counterpart for online exploration. Interestingly, Bellman rank depends both on the environmental dynamics *and the function class $\mathcal{F}$*, and in some cases, the latter dependence is crucial to obtaining low-rankness (e.g., for Linear Quadratic Regulators; see their Proposition 5). Similarly, we may improve the definition of concentratability and make it more widely applicable by incorporating $\mathcal{F}$ into the definition. In Appendix G, we discuss some preliminary ideas based on the theoretical results in this paper.

Acknowledgements

We gratefully thank the constructive comments from Alekh Agarwal and Anonymous Reviewer #3.

References

Abel, D., Hershkowitz, D. E., and Littman, M. L. Near optimal behavior via approximate state abstraction. In *Proceedings of the 33rd International Conference on International Conference on Machine Learning-Volume 48*, pp. 2915–2923. JMLR. org, 2016.

Agrawal, S. *IEOR 8100: Reinforcement Learning. Lecture 4: Approximate Dynamic Programming*. Columbia University, 2018. https://ieor8100.github.io/rl/docs/Lecture%204%20-%20approximate%20DP.pdf.

Antos, A., Szepesvári, C., and Munos, R. Learning near-optimal policies with bellman-residual minimization based fitted policy iteration and a single sample path. *Machine Learning*, 71(1):89–129, 2008.

Auer, P., Cesa-Bianchi, N., and Fischer, P. Finite-time analysis of the multiarmed bandit problem. *Machine learning*, 47(2-3):235–256, 2002.

Bertsekas, D. P. and Tsitsiklis, J. N. *Neuro-Dynamic Programming*. Athena Scientific, Belmont, MA, 1996.

Chen, Y., Li, L., and Wang, M. Scalable bilinear π learning using state and action features. *arXiv preprint arXiv:1804.10328*, 2018.

Dai, B., Shaw, A., Li, L., Xiao, L., He, N., Liu, Z., Chen, J., and Song, L. Sbeed: Convergent reinforcement learning with nonlinear function approximation. In *International Conference on Machine Learning*, pp. 1133–1142, 2018.

Dann, C., Jiang, N., Krishnamurthy, A., Agarwal, A., Langford, J., and Schapire, R. E. On Oracle-Efficient PAC RL with Rich Observations. In *Advances in Neural Information Processing Systems*, pp. 1429–1439, 2018.

Ernst, D., Geurts, P., and Wehenkel, L. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6:503–556, 2005.

Even-Dar, E. and Mansour, Y. Approximate equivalence of Markov decision processes. In *Learning Theory and Kernel Machines*, pp. 581–594. 2003.

Farahmand, A.-m. Regularization in reinforcement learning. 2011.

Farahmand, A.-m., Szepesvári, C., and Munos, R. Error Propagation for Approximate Policy and Value Iteration. In *Advances in Neural Information Processing Systems*, pp. 568–576, 2010.

Farahmand, A.-m., Ghavamzadeh, M., Szepesvári, C., and Mannor, S. Regularized policy iteration with nonparametric function spaces. *The Journal of Machine Learning Research*, 17(1):4809–4874, 2016.

Gordon, G. J. Stable function approximation in dynamic programming. In *Proceedings of the twelfth international conference on machine learning*, pp. 261–268, 1995.

Hutter, M. Extreme state aggregation beyond mdps. In *International Conference on Algorithmic Learning Theory*, pp. 185–199. Springer, 2014.

Jiang, N. *CS 598: Notes on State Abstractions*. University of Illinois at Urbana-Champaign, 2018. http://nanjiang.cs.illinois.edu/files/cs598/note4.pdf.

Jiang, N., Kulesza, A., and Singh, S. Abstraction Selection in Model-based Reinforcement Learning. In *Proceedings of the 32nd International Conference on Machine Learning*, pp. 179–188, 2015.

Jiang, N., Krishnamurthy, A., Agarwal, A., Langford, J., and Schapire, R. E. Contextual Decision Processes with low Bellman rank are PAC-learnable. In *International Conference on Machine Learning*, 2017.

Johnson, M., Hofmann, K., Hutton, T., and Bignell, D. The malmo platform for artificial intelligence experimentation. In *International joint conference on artificial intelligence (IJCAI)*, pp. 4246, 2016.

Kakade, S. and Langford, J. Approximately Optimal Approximate Reinforcement Learning. In *Proceedings of the 19th International Conference on Machine Learning*, volume 2, pp. 267–274, 2002.

Kakade, S. M. *On the sample complexity of reinforcement learning*. PhD thesis, University of College London, 2003.

Krishnamurthy, A., Agarwal, A., and Langford, J. PAC reinforcement learning with rich observations. In *Advances in Neural Information Processing Systems*, pp. 1840–1848, 2016.

Lagoudakis, M. G. and Parr, R. Least-squares policy iteration. *The Journal of Machine Learning Research*, 4: 1107–1149, 2003.

Lazaric, A., Ghavamzadeh, M., and Munos, R. Finite-sample analysis of least-squares policy iteration. *The Journal of Machine Learning Research*, 13(1):3041–3074, 2012.

Li, L. *A unifying framework for computational reinforcement learning theory*. PhD thesis, Rutgers, The State University of New Jersey, 2009.

Li, L., Walsh, T. J., and Littman, M. L. Towards a unified theory of state abstraction for MDPs. In *Proceedings of the 9th International Symposium on Artificial Intelligence and Mathematics*, pp. 531–539, 2006.

Liu, Q., Li, L., Tang, Z., and Zhou, D. Breaking the curse of horizon: Infinite-horizon off-policy estimation. In *Advances in Neural Information Processing Systems*, pp. 5361–5371, 2018.

Lizotte, D. J. Convergent fitted value iteration with linear function approximation. In *Advances in Neural Information Processing Systems*, pp. 2537–2545, 2011.

Lu, T., Schuurmans, D., and Boutilier, C. Non-delusional q-learning and value-iteration. In *Advances in Neural Information Processing Systems*, pp. 9971–9981, 2018.

Maillard, O.-A., Munos, R., Lazaric, A., and Ghavamzadeh, M. Finite-sample analysis of bellman residual minimization. In *Proceedings of 2nd Asian Conference on Machine Learning*, pp. 299–314, 2010.

Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540): 529–533, 2015.

Müller, A. Integral probability metrics and their generating classes of functions. *Advances in Applied Probability*, 29 (2):429–443, 1997.

Munos, R. Error bounds for approximate policy iteration. In *ICML*, volume 3, pp. 560–567, 2003.

Munos, R. Performance bounds in l_p-norm for approximate value iteration. *SIAM journal on control and optimization*, 46(2):541–561, 2007.

Munos, R. and Szepesvári, C. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9 (May):815–857, 2008.

Pires, B. A. and Szepesvári, C. Statistical linear estimation with penalized estimators: an application to reinforcement learning. *arXiv preprint arXiv:1206.6444*, 2012.

Pires, B. Á. and Szepesvári, C. Policy error bounds for model-based reinforcement learning with factored linear models. In *Conference on Learning Theory*, pp. 121–151, 2016.

Ravindran, B. *An algebraic approach to abstraction in reinforcement learning*. PhD thesis, University of Massachusetts Amherst, 2004.

Riedmiller, M. Neural fitted q iteration–first experiences with a data efficient neural reinforcement learning method. In *European Conference on Machine Learning*, pp. 317–328. Springer, 2005.

Singh, S. and Yee, R. An upper bound on the loss from approximate optimal-value functions. *Machine Learning*, 16(3):227–233, 1994.

Sun, W., Jiang, N., Krishnamurthy, A., Agarwal, A., and Langford, J. Model-based RL in Contextual Decision Processes: PAC bounds and Exponential Improvements over Model-free Approaches. In *Conference on Learning Theory*, 2019.

Sutton, R. Introduction to reinforcement learning with function approximation. In *Tutorial at the Conference on Neural Information Processing Systems*, 2015.

Sutton, R. S. and Barto, A. G. *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, MA, March 1998. ISBN 0-262-19398-1.

Sutton, R. S. and Barto, A. G. *Reinforcement learning: An introduction*. MIT press, 2018.

Szepesvári, C. Algorithms for reinforcement learning. *Synthesis lectures on artificial intelligence and machine learning*, 4(1):1–103, 2010.

Szepesvári, C. and Munos, R. Finite time bounds for sampling based fitted value iteration. In *Proceedings of the 22nd international conference on Machine learning*, pp. 880–887. ACM, 2005.

Szepesvári, C. and Smart, W. D. Interpolation-based q-learning. In *Proceedings of the twenty-first international conference on Machine learning*, pp. 100. ACM, 2004.

Tosatto, S., Pirotta, M., D'Eramo, C., and Restelli, M. Boosted fitted q-iteration. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pp. 3434–3443. JMLR. org, 2017.

Tsitsiklis, J. N. and Van Roy, B. An analysis of temporal-difference learning with function approximation. *IEEE TRANSACTIONS ON AUTOMATIC CONTROL*, 42(5), 1997.

Van Hasselt, H., Doron, Y., Strub, F., Hessel, M., Sonnerat, N., and Modayil, J. Deep reinforcement learning and the deadly triad. *arXiv preprint arXiv:1812.02648*, 2018.

Van Roy, B. *Feature-based methods for large scale dynamic programming*. PhD thesis, Massachusetts Institute of Technology, 1994.

Whitt, W. Approximations of dynamic programs, I. *Mathematics of Operations Research*, 3(3):231–243, 1978.

Yang, Z., Xie, Y., and Wang, Z. A Theoretical Analysis of Deep Q-Learning. *arXiv preprint arXiv:1901.00137*, 2019.

Yu, B. Assouad, Fano, and Le Cam. In *Festschrift for Lucien Le Cam*, pp. 423–435. Springer, 1997.