

On Value Functions and the Agent-Environment Boundary

Nan Jiang

Department of Computer Science
University of Illinois at Urbana-Champaign
Urbana, IL 61801
nanjiang@illinois.edu

Abstract

When function approximation is deployed in reinforcement learning (RL), the same problem may be formulated in different ways, often by treating a pre-processing step as a part of the environment or as part of the agent. As a consequence, fundamental concepts in RL, such as (optimal) value functions, are not uniquely defined as they depend on where we draw this *agent-environment boundary*, causing problems in theoretical analyses that provide optimality guarantees. We address this issue via a simple and novel *boundary-invariant* analysis of Fitted Q-Iteration, a representative RL algorithm, where the assumptions and the guarantees are invariant to the choice of boundary. We also discuss closely related issues on state resetting and Monte-Carlo Tree Search, deterministic vs stochastic systems, imitation learning, and the verifiability of theoretical assumptions from data.

1 Introduction

The entire theory of RL—including that of function approximation—is built on mathematical concepts established in the Markov Decision Process (MDP) literature [Puterman, 1994], such as the optimal state- and Q -value functions (V^* and Q^*) and their policy-specific counterparts (V^π and Q^π). These functions operate on the state (and action) of the MDP, and classical results tell us that they are always uniquely and well defined.

Are they really well defined?

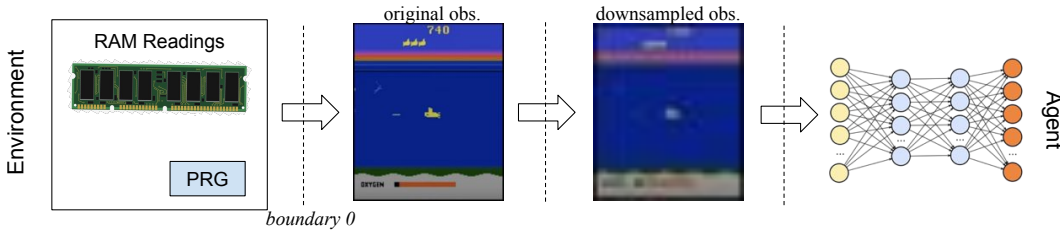


Figure 1: Illustration of the agent-environment boundaries.¹

Consider the following scenario, depicted in Figure 1. In a standard ALE benchmark [Bellemare et al., 2013], raw-pixel screens are produced as states (or strictly speaking, observations¹), and the agent

¹In Atari games, it is common to include past frames in the state representation to resolve partial observability, which is omitted in Figure 1. In most part of the paper we stick to MDP terminologies for simplicity, but our results and discussions also apply to POMDPs. See Appendix B for details.

feeds the state into a neural net to predict Q^* . Since the original game screen has a high resolution, it is common in practice to downsample the screen as a pre-processing step [Mnih et al., 2015].

There are two equivalent views of this scenario: We can either view the pre-processing step as part of the environment, or as part of the agent. Depending on where we draw this *agent-environment boundary*, Q^* will be different in general.² It should also be obvious that there may exist many choices of the boundary (e.g., “*boundary 0*” in Figure 1), some of which we may be even not aware of. When we design an algorithm to learn Q^* , which Q^* are we talking about?

The good news is that many existing algorithms (with exceptions; see Section 3) are *boundary-invariant*, that is, the behavior of the algorithm remains the same however we change the boundary. The bad news is that many existing analyses³ are *boundary-dependent*, as they make assumptions that may either hold or fail in the same problem depending on the choice of the boundary; for example, in the analyses of approximate value iteration algorithms, it is common to assume that Q^* can be represented by the function approximator (“*realizability*”), and that the function space is closed under Bellman update [“*low inherent Bellman error*”, Szepesvári and Munos, 2005, Antos et al., 2008]. Such a gap between the mathematical theory and the reality also leads to further consequences, such as the theoretical assumptions being fundamentally unverifiable from naturally generated data.

In this paper we systematically study the boundary dependence of RL theory. We ground our discussions in a simple and novel *boundary-invariant* analysis of Fitted Q-Iteration [Ernst et al., 2005], in which the correctness of the assumptions and the guarantees do not change with the subjective choice of the boundary (Sections 4 and 5). Within this analysis, we give up on the classical notions of value functions or even the (state-wise) Bellman equation, and replace them with weaker conditions that are boundary-invariant and that naturally come with improved verifiability. We also discuss closely related issues on state resetting and Monte-Carlo Tree Search (Section 3), deterministic vs stochastic systems, imitation learning, and the verifiability of theoretical assumptions from data (Section 6).

2 Preliminaries

Markov Decision Processes An infinite-horizon discounted MDP M is specified by $(\mathcal{S}, \mathcal{A}, P, R, \gamma, d_0)$, where \mathcal{S} is the finite state space,⁴ \mathcal{A} is the finite action space, $P : \mathcal{S} \times \mathcal{A} \rightarrow \Delta(\mathcal{S})$ is the transition function,⁵ $R : \mathcal{S} \times \mathcal{A} \rightarrow \Delta([0, R_{\max}])$ is the reward function, $\gamma \in [0, 1)$ is the discount factor, and $d_0 \in \Delta(\mathcal{S})$ is the initial state distribution.

A (stationary and deterministic) policy $\pi : \mathcal{S} \rightarrow \mathcal{A}$ specifies a decision-making strategy, and induces a distribution over random trajectories: $s_1 \sim d_0$, $a_1 \sim \pi$, $r_1 \sim R(s_1, a_1)$, $s_2 \sim P(s_1, a_1)$, $a_2 \sim \pi$, ..., where $a_t \sim \pi$ is short for $a_t = \pi(s_t)$. In later analyses, we will also consider stochastic policies $\pi : \mathcal{S} \rightarrow \Delta(\mathcal{A})$ and non-stationary policies formed by concatenating a sequence of stationary ones.

The performance of a policy π is measured by its expected discounted return (or value):⁶

$$v^\pi := \mathbb{E}[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \mid s_1 \sim d_0, \pi].$$

The value of a policy lies in the range of $[0, V_{\max}]$ with $V_{\max} = R_{\max}/(1 - \gamma)$. It will be useful to define the Q -value function of π : $Q^\pi(s, a) := \mathbb{E}[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \mid s_1 = s, a_1 = a, a_{2:\infty} \sim \pi]$, and d_t^π , the distribution over state-action pairs induced at time step t : $d_t^\pi(s, a) := \Pr[s_t = s, a_t = a \mid s_1 \sim d_0, \pi]$. Note that $d_1^\pi = d_0 \times \pi$, which means $s \sim d_0, a \sim \pi$.

The goal of the agent is to find a policy π that maximizes v^π . In the infinite-horizon discounted setting, there always exists an optimal policy π^* that maximizes the expected discounted return for all states simultaneously (and hence also for d_0). Let Q^* be a shorthand for Q^{π^*} . It is known that π^* is the greedy policy w.r.t. Q^* : For any Q -function f , let π_f denote its greedy policy ($s \mapsto$

²For concreteness, we provide a minimal example of boundary dependence in Appendix A.

³There are different kinds of theoretical analyses in RL (e.g., convergence analysis). In this paper we focus on analyses that provide near-optimality guarantees.

⁴For the ease of exposition we assume finite \mathcal{S} , but its cardinality can be arbitrarily large.

⁵ $\Delta(\cdot)$ is the probability simplex.

⁶It is important that the performance of a policy is measured under the initial state distribution. See Appendix F for further discussions.

$\arg \max_{a \in \mathcal{A}} f(s, a)$, and we have $\pi^* = \pi_{Q^*}$. Furthermore, Q^* satisfies the Bellman equation: $Q^* = \mathcal{T}Q^*$, where $\mathcal{T} : \mathbb{R}^{\mathcal{S} \times \mathcal{A}} \rightarrow \mathbb{R}^{\mathcal{S} \times \mathcal{A}}$ is the Bellman optimality operator:

$$(\mathcal{T}f)(s, a) = \mathbb{E}_{r \sim R(s, a), s' \sim P(s, a)} [r + \gamma \max_{a' \in \mathcal{A}} f(s', a')]. \quad (1)$$

Value-function Approximation In complex problems with high-dimensional observations, function approximation is often deployed to generalize over the large state space. In this paper we take a learning-theoretic view of value-function approximation: We are given a function space $\mathcal{F} \subset (\mathcal{S} \times \mathcal{A} \rightarrow [0, V_{\max}])$, and for simplicity we assume \mathcal{F} is finite⁷. The goal—stated in the classical, boundary-dependent fashion—is to identify a function $f \in \mathcal{F}$ such that $f \approx Q^*$, so that π_f is a near-optimal policy. This naturally motivates a common assumption, known as *realizability*, that $Q^* \in \mathcal{F}$, which will be useful for our discussions in Section 3.

For most of the paper we will be concerned with batch-mode value-function approximation, that is, the learner is passively given a dataset $\{(s, a, r, s')\}$ and cannot directly interact with the environment. The implications in the exploration setting will be briefly discussed at the end of the paper.

Fitted Q-Iteration (FQI) FQI [Ernst et al., 2005, Szepesvári, 2010] is a batch RL algorithm that solves a sequence of least-squared regression problems with \mathcal{F} to approximate each step of value iteration. It is also considered as the prototype for the popular DQN algorithm [Mnih et al., 2015], and often used as a representative of off-policy value-based RL algorithms in empirical studies [Fu et al., 2019]. We defer a detailed description of the algorithm to Section 5.1.

3 On the Nonuniqueness of Q^*

In this section we expand the discussions in Section 1 and introduce a few interesting paradoxes and questions to develop a deeper understanding of the agent-environment boundary, and to motivate our later analyses with theoretical and practical concerns. We will leave some questions open and revisit them after the technical sections.

Paradox: Q^* Uniquely Defined via State Resetting?

While the boundary dependence of Q^* should be intuitive from Section 1, one might find a conflict in the following fact: In complex simulated RL environments, we often approximate the Q^* via Monte-Carlo Tree Search [Kearns et al., 2002, Kocsis and Szepesvári, 2006], either for the purpose of generating expert demonstration for imitation learning [Guo et al., 2014] or for code debugging and sanity check. For example, one can collect a regression dataset $\{(s, a), Q^*(s, a)\}$ where $Q^*(s, a)$ is approximated by MCTS, and verify whether $Q^* \in \mathcal{F}$ by solving the regression problem over \mathcal{F} . MCTS enjoys near-optimality guarantees without using any form of function approximation, so how can Q^* not be well-defined?

The answer lies in the way that MCTS interacts with the simulator: At each time step, MCTS rolls out multiple trajectories from the *current state* to determine the optimal action, often by *resetting the state* after each simulation trajectory is completed. There are many ways of resetting the state: For example, one can clone the RAM configuration of the “real” state and always reset to that (“*boundary 0*” in Figure 1), as done by Guo et al. [2014]. One can also attempt to reproduce the sequence of observations and actions from the beginning of the episode [see POMCP; Silver and Veness, 2010]. Both are valid state resetting operations but for different choices of the boundary.^{8,9}

Practical Concern The above discussion reveals a practical concern, that when we compute Q^* (via MCTS or other methods) for sanity checking the realizability of \mathcal{F} , we have to explicitly choose a boundary, which may or may not be the best choice for the given \mathcal{F} . More importantly, not all RL problems come with the resetting functionality, and without resetting it is fundamentally impossible to check certain assumptions, such as realizability; see Appendix C for a formal argument and proof.

⁷The only reason that we assume finite \mathcal{F} is for mathematical convenience in Theorem 3, and removing this assumption only has minor impact on our results. See further comments after Theorem 3’s proof in Appendix E.

⁸This is to say, MCTS is a family of boundary-dependent algorithms, and they can only have boundary-dependent analyses. (The problem with them is how to pick the boundary as a design choice; see Section 6.) In contrast, most algorithms that do not require state resetting are boundary-invariant hence should have boundary-invariant analyses, which we derive in this paper.

⁹Another natural boundary corresponds to resetting the contents of the RAM but leaving the PRG state intact.

Question 1. *When the appropriate boundary is unclear or state resetting is not available, how to empirically verify the theoretical assumptions?*

Theoretical Concern Relatedly, since the validity of the common assumptions and the guarantees generally depend on the boundary, given an arbitrary MDP M and a function class \mathcal{F} , we may naturally ask if re-expressing the problem as some equivalent (M', \mathcal{F}') may result in better guarantees, and what is the “best boundary” for stating the theoretical guarantees for a problem and how to characterize it. Intuitions tell us that we might want to choose the “rightmost” boundary (direction defined according to Figure 1), that is, all pre-processing steps should belong to the environment. This corresponds to the state compression scheme, $s \mapsto \{f(s, a) : a \in \mathcal{A}, f \in \mathcal{F}\}$ [Sun et al., 2019], but such a definition is very brittle, as even the slightest numerical changes in \mathcal{F} might cause the boundary to change significantly.¹⁰

Question 2. *How to robustly define the boundary that provides the best theoretical guarantees?*

Solution: Boundary-Invariant Analyses We answer the two questions and address the practical and theoretical concerns via a novel *boundary-invariant* analysis, which we derive in the following sections. While it is impossible to cover all existing algorithms, we exemplify the analysis for a representative value-based algorithm (FQI), and believe that the spirit and the techniques are widely applicable. For FQI, we show that it is possible to relax the common assumptions in literature to their boundary-invariant counterparts and still provide the same near-optimality guarantees. This addresses Question 2, as we provide a compelling guarantee compared to that of classical boundary-dependent analyses under *any* boundary, so there is no need to choose a boundary whatsoever. Our assumptions also have improved verifiability than their boundary-dependent counterparts, which partially addresses Question 1.

4 Case Study: Batch Contextual Bandit (CB) with Predictable Rewards

We warm-start with the simple problem of fitting a reward function from batch data in contextual bandits, which may be viewed as MDPs with $\gamma = 0$.¹¹ The analysis also applies straightforwardly to learning a policy-specific value-function Q^π from Monte-Carlo rollouts and performing one-step policy improvement [Sutton and Barto, 2018]. This section also provides important building blocks for Section 5, and the simplicity of the analysis allows us to thoroughly discuss the intuitions and the conceptual issues, leaving Section 5 focused on the technical aspects.

4.1 Setting and Algorithm

Let $D = \{(s, a, r)\}$ be a dataset, where $s \sim d_0$, $a \sim \pi_b$ and $r \sim R(s, a)$. Let μ denote the joint distribution over (s, a) , or $\mu := d_0 \times \pi_b$. For any $f \in \mathcal{F}$, define the empirical squared loss

$$\mathcal{L}_D(f) := \frac{1}{|D|} \sum_{(s,a,r) \in D} (f(s, a) - r)^2, \quad (2)$$

and the population version $\mathcal{L}_\mu(f) := \mathbb{E}_D[\mathcal{L}_D(f)]$. The algorithm fits a reward function by minimizing $\mathcal{L}_D(\cdot)$, that is, $\hat{f} := \arg \min_{f \in \mathcal{F}} \mathcal{L}_D(f)$, and outputs $\pi_{\hat{f}}$. We are interested in providing a guarantee to the performance of this policy, that is, the expected reward obtained by executing $\pi_{\hat{f}}$, $v^{\pi_{\hat{f}}} = \mathbb{E}[r \mid \pi_{\hat{f}}]$. We will base our analyses on the following inequality:

$$\mathcal{L}_\mu(\hat{f}) - \min_{f \in \mathcal{F}} \mathcal{L}_\mu(f) \leq \epsilon. \quad (3)$$

In words, we assume that \hat{f} approximately minimizes the population loss. Such a bound can be obtained via a uniform convergence argument, where ϵ will depend on the sample size $|D|$ and the statistical complexity of \mathcal{F} (e.g., pseudo-dimension [Haussler, 1992]). We do not include this part as it is standard and orthogonal to the discussions in this paper, and rather focus on how to provide a guarantee on $v^{\pi_{\hat{f}}}$ as a function ϵ .

¹⁰If we take an arbitrary function and add arbitrarily small perturbations to $f(s, a)$ for each (s, a) , $\{f(s, a) : a \in \mathcal{A}, f \in \mathcal{F}\}$ might reveal all the information in s and the mapping is essentially an isomorphism.

¹¹In Sections 4 and 5, the same symbol carries the same (or similar) meaning. However, there are some inevitable differences and the reader should not confuse the two settings in general.

4.2 A Sufficient Condition for Boundary Invariance

Before we start the analysis, we first show that the algorithm itself is boundary-invariant, leading to a sufficient condition for judging the boundary invariance of analyses. The concept central to the algorithm is the squared loss $\mathcal{L}_D(f)$. Although $\mathcal{L}_D(f)$ is defined using (s, a) , the definition refers to (s, a) exclusively through the evaluation of $f \in \mathcal{F}$ on (s, a) , taking expectation over a naturally generated dataset D .¹² The data points are generated by an objective procedure (collecting data with policy π_b), and on every data point (s, a, r) , $f(s, a)$ is the same scalar regardless of the boundary, hence the algorithmic procedure is boundary-invariant.

Inspired by this, we provide the following sufficient condition for boundary-invariant analyses:

Claim 1. *An analysis is boundary-invariant if the assumptions and the optimal value are defined in a way that accesses states and actions exclusively through evaluations of functions in \mathcal{F} , with plain expectations (either empirical or population) over naturally generated data distributions.*

A number of pitfalls need to be avoided in the specification of such a condition:

- Restricting the functions to \mathcal{F} is important, as one can define conditional expectations (on a single state) through plain expectations via the use of state indicator functions (or the dirac delta functions for continuous state spaces).
- Similarly, we need to restrict the set of distributions to those natural ones (formalized later in Definition 2) to prevent expectations on point masses. (See Appendix F for further discussions.)
- Besides the assumptions, the very notion of optimality also needs to be taken care of, as $v^* := \mathbb{E}_{s \sim d_0}[V^*(s)]$ (the usual notion of optimal value) is also a boundary-dependent quantity.¹³

That said, this condition is not perfectly rigorous, as we find it difficult to make it mathematically strict without being verbose and/or restrictive. Regardless, we believe it conveys the right intuitions and can serve as a useful guideline for judging the boundary invariance of a theory. Furthermore, the condition provides us with significant mathematical convenience: as long as the condition is satisfied, we can analyze an algorithm under *any* boundary, allowing us to use the standard MDP formulations and all the objects defined therein (states, actions, their distributions, etc.).

4.3 Classical Assumptions

We now review the classical assumptions in this problem for later references and comparisons. The first assumption is that data is exploratory, often guaranteed by taking randomized actions in the data collection policy (or behavior policy π_b) and not starving any of the actions:

Assumption 1 (π_b is exploratory). There exists a universal constant $C < +\infty$ such that,
 $\forall s \in \mathcal{S}, a \in \mathcal{A}, \quad \pi_b(a|s) \geq 1/C$.

The second one is the realizability as already discussed in Section 2.

Assumption 2 (Realizability). $Q^* \in \mathcal{F}$. In contextual bandits, $Q^*(s, a) = \mathbb{E}[r | s, a], \forall (s, a)$.

Two comments before we move on:

- While we consider exact realizability for simplicity, it is possible to allow an approximation error and state a guarantee that degrades gracefully with the violation of the assumption. Such an extension is routine and we omit it for readability.
- We do not provide the classical analysis here, and simply note that the guarantee is the same as Theorem 2 when Assumptions 1 and 2 hold.

4.4 Boundary-Invariant Assumptions and Analysis

Additional Notations For any $f : \mathbb{R}^{\mathcal{S} \times \mathcal{A}}$ and any distribution $\nu \in \Delta(\mathcal{S} \times \mathcal{A})$, define $\|f\|_\nu^2 := \mathbb{E}_{(s,a) \sim \nu}[f(s,a)^2]$. To improve readability we will often omit “ $r \sim R(s, a)$ ” when an expectation involves r (and “ $s' \sim P(s, a)$ ” for Section 5), and use $\mathbb{E}_\nu[f]$ as a shorthand for $\mathbb{E}_{(s,a) \sim \nu}[f(s, a)]$.

¹²Here we use “naturally generated” to contrast state resetting operations discussed in Section 3.

¹³For any fixed π , v^π is boundary-invariant. Here the boundary dependence of v^* is due to that of π^* .

Definition 1 (Admissible distributions (bandit)). Given a contextual bandit problem and a space of candidate reward functions \mathcal{F} , we call $\{d_0 \times \pi_f : f \in \mathcal{F}\}$ the space of *admissible distributions*.

Assumption 3. There exists a universal constant $C < +\infty$ such that, for any $f, f' \in \mathcal{F}$ and any admissible ν , $\|f - f'\|_\mu^2 \leq C \cdot \|f - f'\|_\nu^2$.

Assumption 3 is a direct consequence of Assumption 1, as C is an upper bound on the ℓ_∞ norm of the importance ratio between ν and μ . The proof is elementary and omitted.

Assumption 4. There exists $f^* \in \mathcal{F}$, such that for all admissible ν ,

$$\mathbb{E}_\nu[f^*] = \mathbb{E}_{(s,a) \sim \nu}[r], \quad (4)$$

and for any $f' \in \mathcal{F}$,

$$\mathcal{L}_\mu(f') - \mathcal{L}_\mu(f^*) = \|f' - f^*\|_\mu^2. \quad (5)$$

We say that such an f^* is a valid reward function of the CB w.r.t. \mathcal{F} .

Assumption 4 is implied by Assumption 2, as $f^* = Q^*$ satisfies both Eq.(4) and Eq.(5): Eq.(4) can be obtained from $Q^*(s, a) = \mathbb{E}[r|s, a]$ by taking the expectation of both sides w.r.t. ν . Eq.(5) is the standard bias-variance decomposition for squared loss regression when f^* is the Bayes-optimal regressor.¹⁴ Eq.(4) guarantees that f^* still bears the semantics of reward, although no longer in a point-wise manner. Eq.(5) guarantees that f^* can be reliably identified through squared loss minimization, which is specialized to the batch learning setting with squared loss minimization. In fact, we provide a counter-example in Appendix D showing that dropping Eq.(5) can result in the failure of the algorithm, and also discuss other learning settings where this assumption is not needed.

Now we are ready to state the main theorem of this section, whose proof can be found in Appendix E.

Theorem 2. Under Assumptions 3 and 4, for any valid f^* , we have $v^{\pi_{\hat{f}}} \geq \mathbb{E}_{d_0 \times \pi_{f^*}}[f^*] - 2\sqrt{C}\epsilon$.

5 Case Study: Fitted Q-Iteration

5.1 Setting and Algorithm

To highlight the differences between boundary-dependent and boundary-invariant analyses, we adopt a simplified setting assuming i.i.d. data. Interested readers can consult prior works for more general analyses on β -mixing data [e.g., Antos et al., 2008].

Let $D = \{(s, a, r, s')\}$ be a dataset, where $(s, a) \sim \mu$, $r \sim R(s, a)$, and $s' \sim P(s, a)$. For any $f, f' \in \mathcal{F}$, define the empirical squared loss

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

and the population version $\mathcal{L}_\mu(f; f') := \mathbb{E}_D[\mathcal{L}_D(f; f')]$. The algorithm initializes f_1 arbitrarily, and

$$f_i := \arg \min_{f \in \mathcal{F}} \mathcal{L}_D(f; f_{i-1}), \quad \text{for } i \geq 2.$$

The algorithm repeats this for some k iterations and outputs π_{f_k} . We are interested in providing a guarantee to the performance of this policy.

5.2 Classical Assumptions

Similar to the CB case, there will be two assumptions, one that requires the data to be exploratory, and one that requires \mathcal{F} to satisfy certain representation conditions.

Definition 2 (Admissible distributions (MDP)). A state-action distribution is admissible if it takes the form of d_t^π for any t , and any non-stationary policy π formed by choosing a policy for each time step from $\{\pi_f : f \in \mathcal{F}\} \cup \{\pi_{f,f'} : f, f' \in \mathcal{F}\}$, where $\pi_{f,f'}(s) := \arg \max_{a \in \mathcal{A}} \max(f(s, a), f'(s, a))$.

Assumption 5 (μ is exploratory). There exists a universal constant $C < +\infty$ such that for any admissible ν , $\max_{s,a} \frac{\nu(s,a)}{\mu(s,a)} \leq C$.

¹⁴It is easy to allow an approximation error in Eq.(4) and/or (5). For example, one can measure the violation of Eq.(4) by $\inf_{f \in \mathcal{F}} \sup_\nu |\mathbb{E}_{(s,a) \sim \nu}[f - r]|$, and such errors can be easily incorporated in our later analysis.

This guarantees that μ well covers all admissible distributions. The upper bound C is known as the *concentratability coefficient* [Munos, 2003], and here we use the simplified version from a recent analysis by Chen and Jiang [2019]. See Farahmand et al. [2010] for a more fine-grained characterization of this quantity.

Assumption 6 (No inherent Bellman error). $\forall f \in \mathcal{F}, \mathcal{T}f \in \mathcal{F}$.

This assumption states that \mathcal{F} is closed under the Bellman update operator \mathcal{T} . It automatically implies $Q^* \in \mathcal{F}$ (for finite \mathcal{F}) hence is stronger than realizability, but replacing this assumption with realizability can cause FQI to diverge [Van Roy, 1994, Gordon, 1995, Tsitsiklis and Van Roy, 1997] or have exponential sample complexity [Dann et al., 2018]. We refer the readers to Chen and Jiang [2019] for further discussions on the necessity of this assumption.

It is also possible to relax the assumption and allow an approximation error in the form of $\sup_{f \in \mathcal{F}} \inf_{f' \in \mathcal{F}} \|f' - \mathcal{T}f\|$, known as the *inherent Bellman error* [Munos and Szepesvári, 2008]. Again we do not consider this extension, and incorporating it in our analysis is straightforward.

5.3 Boundary-Invariant Assumptions

We give the boundary-invariant counterparts of Assumptions 5 and 6.

Assumption 7. There exists a universal constant $C < +\infty$ such that, for any $f, f' \in \mathcal{F}$ and any admissible state-action distribution ν , $\|f - f'\|_\nu^2 \leq C \|f - f'\|_\mu^2$.

Assumption 8. $\forall f \in \mathcal{F}$, there exists $g \in \mathcal{F}$ such that for all admissible ν ,

$$\mathbb{E}_\nu[g] = \mathbb{E}_{(s,a) \sim \nu}[r + \gamma \max_{a' \in \mathcal{A}} f(s', a')], \quad (6)$$

and for any $f' \in \mathcal{F}$,

$$\mathcal{L}_\mu(f'; f) - \mathcal{L}_\mu(g; f) = \|f' - g\|_\mu^2. \quad (7)$$

Define \mathcal{B} as the operator that maps f to an arbitrary (but systematically chosen) g that satisfies the above conditions.

Assumption 8 states that for every $f \in \mathcal{F}$, we can define a contextual bandit problem with random reward $r + \gamma \max_{a' \in \mathcal{A}} f(s', a')$, and there exists $g \in \mathcal{F}$ that is a valid reward function for this problem (Assumption 4). In the classical definitions, the true reward function for this problem is $\mathcal{T}f$, so our \mathcal{B} operator can be viewed as the boundary-invariant version of \mathcal{T} .

5.4 Boundary-Invariant Analysis

In Section 4 for contextual bandits, f^* is defined directly in the assumptions, and we use it to define the optimal value in Theorem 2. In Assumptions 7 and 8, however, no counterpart of Q^* is defined. How do we even express the optimal value that we compete with?

We resolve this difficulty by relying on the \mathcal{B} operator defined in Assumption 8. Recall that in classical analyses, Q^* can be defined as the fixed point of \mathcal{T} , so we define f^* similarly through \mathcal{B} .

Theorem 3. Under Assumption 8, there exists $f^* \in \mathcal{F}$ s.t. $\|\mathcal{B}f^* - f^*\|_\nu = 0$ for any admissible ν .

The key to proving Theorem 3 is to show a γ -contraction-like property of \mathcal{B} , formalized in Lemma 4.

Lemma 4 (Boundary-invariant version of γ -contraction). Under Assumption 8, for any admissible ν , $\forall f, f' \in \mathcal{F}$, let $\pi_{f,f'}(s) := \arg \max_{a \in \mathcal{A}} \max(f(s, a), f'(s, a))$, and $P(\nu)$ denote the distribution of s' generated as $(s, a) \sim \nu, s' \sim P(s, a)$,

$$\|\mathcal{B}f - \mathcal{B}f'\|_\nu \leq \gamma \|f - f'\|_{P(\nu) \times \pi_{f,f'}}. \quad (8)$$

Although similar results are also proved in classical analyses, proving Lemma 4 under Assumption 8 is more challenging. For example, a very useful property in the classical analysis is that $\mathbb{E}[(\mathcal{T}f)(s, a) - r - \gamma \max_{a'} f(s', a') | s, a] = 0$, and it holds in a point-wise manner for every (s, a) . In our boundary-invariant analyses, however, such a handy tool is not available as we only make assumptions on the average-case properties of the functions, and their point-wise behavior is undefined. We refer the readers to Appendix E for how we overcome this technical difficulty.

With f^* defined in Theorem 3, we state the main theorem of this section, with proof deferred to Appendix E.

Theorem 5. Let $f_1, f_2, \dots, f_k \in \mathcal{F}$ be the sequence of functions obtained by FQI. Let ϵ be an universal upper bound on the error incurred in each iteration, that is, $\forall 1 \leq i \leq k-1$,

$$\mathcal{L}_\mu(f_k; f_{k-1}) \leq \min_{f \in \mathcal{F}} \mathcal{L}_\mu(f; f_{k-1}) + \epsilon.$$

Let $\hat{\pi}$ be the greedy policy of f_k . Then $v^{\hat{\pi}} \geq v^{\pi_{f^*}} - \frac{2}{1-\gamma}(\frac{\sqrt{C}\epsilon}{1-\gamma} + \gamma^k V_{\max})$.

6 Discussions

We conclude the paper with further discussions and open questions.

Verifiability The ability to verify the correctness of theoretical assumptions is important to the development and the debugging of RL algorithms. In Section 3 we argued that classical realizability-type assumptions are not only boundary-dependent, but also cannot be verified from naturally generated data without state resetting. One major difficulty is that quantities like Q^* are defined via conditional expectations “ $\mathbb{E}[\cdot \mid s_1 = s, a_1 = a]$ ”, and estimating it requires reproducing the same state multiple times, which is impossible in general. This issue is eliminated in the boundary-invariant analyses, as the assumptions are stated using plain expectations over data (recall Claim 1), which can be verified (up to any accuracy) via Monte-Carlo estimation.

Of course, verifying the boundary-invariant assumptions still faces significant challenges, as the statements frequently use languages like “ $\forall f \in \mathcal{F}$ ” and “ \forall admissible ν ”, making it computationally expensive to verify them exhaustively. We note, however, that this is likely to be the case for any strict theoretical assumptions, and practitioners often develop heuristics under the guidance of theory to make the verification process tractable. For example, the difficulty related to “ $\forall f \in \mathcal{F}$ ” may be resolved by clever optimization techniques, and that related to “ $\forall \nu$ ” may be addressed by testing the assumptions on a diverse and representative set of distributions designed with domain knowledge. We leave the design of an efficient and effective verification protocol to future work.

“Boundary 0” As we hinted in Figure 1, there exists a choice of the boundary that makes every RL problem deterministic [Ng and Jordan, 2000]. This leads to a number of further paradoxes: for example, many difficulties in RL arise due to stochastic transitions, and there are algorithms designed for deterministic systems that avoid these difficulties. Why don’t we always use them since all environments are essentially deterministic? This question, among others, is discussed in Appendix F. In general, we find that investigating this extreme view is helpful in clarifying some of the confusions, and it provides justifications for certain design choices in our theory.

Should we discard boundary-dependent analyses? We do not advocate for replacing boundary-dependent analyses with their boundary-invariant counterparts and this is not the intention of this paper. Rather, our purpose is to demonstrate the feasibility of boundary-invariant analyses, and to use the concrete maths to ground the discussions of the conceptual issues (which can easily go astray and become vacuous given the nature of this topic). On a related note, boundary-dependent analyses make stronger assumptions hence are mathematically easier to work with in general.

Boundary invariance in exploration algorithms Boundary-invariant version of Bellman equation for policy evaluation has appeared in Jiang et al. [2017] who study PAC exploration under function approximation, although they do not discuss its further implications. While our assumptions are inspired by theirs, we have to deal with additional technical difficulties due to off-policy policy optimization. In Appendix D we discuss the connections and the differences between the two papers on a concrete example.

MCTS meets value-function approximation In Section 3 we show that the issue of boundary dependence is not just conceptual puzzles and can have real consequences, especially when MCTS and value-function approximation appear together. One can further ask: When we use MCTS to provide expert demonstration for a value-based learner [e.g., Guo et al., 2014], how should we choose the boundary (i.e., which notion of state should we reset to in MCTS)? (The same question can be also asked about the residual algorithms [Baird, 1995], which minimize Bellman errors via the *double sampling* trick, i.e., drawing two i.i.d. next-states s' from the same (s, a) .) More generally, when the learner is of limited capability in an imitation learning scenario [Ross et al., 2011, Ross and Bagnell, 2014], how to best design the demonstration policy? In fact, we show in Appendix G that demonstration using π^* for a poorly chosen boundary can be *completely* useless. Answering these questions is beyond the scope of this paper, and we leave the investigation to future work.

References

- 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.
- Leemon Baird. Residual algorithms: Reinforcement learning with function approximation. In *Machine Learning Proceedings 1995*, pages 30–37. Elsevier, 1995.
- Marc G Bellemare, Yavar Naddaf, Joel Veness, and Michael Bowling. The arcade learning environment: An evaluation platform for general agents. *Journal of Artificial Intelligence Research*, 47:253–279, 2013.
- Jinglin Chen and Nan Jiang. Information-theoretic considerations in batch reinforcement learning. In *Proceedings of the 36th International Conference on Machine Learning*, pages 1042–1051, 2019.
- Christoph Dann, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, John Langford, and Robert E Schapire. On Oracle-Efficient PAC RL with Rich Observations. In *Advances in Neural Information Processing Systems*, pages 1429–1439, 2018.
- Damien Ernst, Pierre Geurts, and Louis Wehenkel. Tree-based batch mode reinforcement learning. *Journal of Machine Learning Research*, 6:503–556, 2005.
- Amir-massoud Farahmand, Csaba Szepesvári, and Rémi Munos. Error Propagation for Approximate Policy and Value Iteration. In *Advances in Neural Information Processing Systems*, pages 568–576, 2010.
- Justin Fu, Aviral Kumar, Matthew Soh, and Sergey Levine. Diagnosing bottlenecks in deep q-learning algorithms. In *Proceedings of the 36th International Conference on Machine Learning*, pages 2021–2030, 2019.
- Geoffrey J Gordon. Stable function approximation in dynamic programming. In *Proceedings of the twelfth international conference on machine learning*, pages 261–268, 1995.
- Xiaoxiao Guo, Satinder Singh, Honglak Lee, Richard L Lewis, and Xiaoshi Wang. Deep learning for real-time atari game play using offline monte-carlo tree search planning. In *Advances in neural information processing systems*, pages 3338–3346, 2014.
- David Haussler. Decision theoretic generalizations of the PAC model for neural net and other learning applications. *Information and computation*, 1992.
- 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*, 2017.
- Sham Kakade and John Langford. Approximately Optimal Approximate Reinforcement Learning. In *Proceedings of the 19th International Conference on Machine Learning*, volume 2, pages 267–274, 2002.
- Michael Kearns, Yishay Mansour, and Andrew Y Ng. A sparse sampling algorithm for near-optimal planning in large Markov decision processes. *Machine Learning*, 49(2-3):193–208, 2002.
- Levente Kocsis and Csaba Szepesvári. Bandit based monte-carlo planning. In *Machine Learning: ECML 2006*, pages 282–293. Springer Berlin Heidelberg, 2006.
- Volodymyr Mnih, Koray Kavukcuoglu, David Silver, Andrei A Rusu, Joel Veness, Marc G Bellemare, Alex Graves, Martin Riedmiller, Andreas K Fidjeland, Georg Ostrovski, et al. Human-level control through deep reinforcement learning. *Nature*, 518(7540):529–533, 2015.
- Rémi Munos. Error bounds for approximate policy iteration. In *ICML*, volume 3, pages 560–567, 2003.
- Rémi Munos and Csaba Szepesvári. Finite-time bounds for fitted value iteration. *Journal of Machine Learning Research*, 9(May):815–857, 2008.

- Andrew Y Ng and Michael Jordan. PEGASUS: A policy search method for large MDPs and POMDPs. In *Proceedings of the Sixteenth conference on Uncertainty in artificial intelligence*, pages 406–415. Morgan Kaufmann Publishers Inc., 2000.
- ML Puterman. Markov Decision Processes. *Jhon Wiley & Sons, New Jersey*, 1994.
- Stephane Ross and J Andrew Bagnell. Reinforcement and imitation learning via interactive no-regret learning. *arXiv preprint arXiv:1406.5979*, 2014.
- Stéphane Ross, Geoffrey Gordon, and Drew Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In *Proceedings of the fourteenth international conference on artificial intelligence and statistics*, pages 627–635, 2011.
- David Silver and Joel Veness. Monte-Carlo planning in large POMDPs. In *Advances in Neural Information Processing Systems*, pages 2164–2172, 2010.
- Wen Sun, Nan Jiang, Akshay Krishnamurthy, Alekh Agarwal, and John Langford. Model-based RL in Contextual Decision Processes: PAC bounds and Exponential Improvements over Model-free Approaches. In *Conference on Learning Theory*, 2019.
- Richard S Sutton and Andrew G Barto. *Reinforcement learning: An introduction*. MIT press, 2018.
- Csaba Szepesvári. Algorithms for reinforcement learning. *Synthesis lectures on artificial intelligence and machine learning*, 4(1):1–103, 2010.
- 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. ACM, 2005.
- John N Tsitsiklis and Benjamin Van Roy. An analysis of temporal-difference learning with function approximation. *IEEE TRANSACTIONS ON AUTOMATIC CONTROL*, 42(5), 1997.
- Benjamin Van Roy. *Feature-based methods for large scale dynamic programming*. PhD thesis, Massachusetts Institute of Technology, 1994.
- David H Wolpert. The lack of a priori distinctions between learning algorithms. *Neural computation*, 8(7):1341–1390, 1996.

A A Minimal Example of Boundary Dependence

For concreteness we provide a minimal example where Q^* changes with the boundary. Consider a contextual bandit problem with two contexts/states, s_A and s_B , and the context distribution is $d_0(s_A) = d_0(s_B) = 0.5$. There is only 1 action, and s_A yields a deterministic reward of 0, and s_B yields a deterministic reward of 1, so $Q^*(s_A) = 0$ and $Q^*(s_B) = 1$. However, if the agent ignores the context, the problem is equivalent to a multi-armed bandit with 1 arm yielding a random reward that is Bernoulli distributed, so predicting 0.5 value for both states gives another valid Q^* function.

B Partial Observability

RL environments are partially observable in general, and even if the original environment is Markovian, partial observability can arise if we treat a lossy/noisy pre-processing step as part of the environment. While we restrict ourselves to MDPs in the main text for the ease of exposition, we note that our results and discussions still apply to the partially observable case. We warn, however, that it is very easy to get confused on this topic, and we provide the following comments to help clarify some of the potential confusions.

1. All POMDPs have well-defined value-functions, as a POMDP can always be treated as an MDP over histories (alternating observations and actions).

2. Let M be the environment without a pre-processing step ϕ , and M_ϕ be the environment that includes ϕ . Discussions in Section 1 are based on the fact that functions that operate on $\phi(s)$ (e.g., the neural net that takes the downsampled images as input in Figure 1) can also be treated as a function of s (e.g., the original image), as $f(\phi(s), a) = (f \circ \phi)(s, a)$.

Now what happens if M is an MDP, but M_ϕ is a POMDP, and functions in \mathcal{F} takes past observations (and actions) as inputs? In this case, f may not be a function of s , as s may have “forgotten” the previous states.

As usual, this can be fixed if we re-express M by treating histories as states (let the new MDP be M'), even if this is redundant as s is Markovian in M . By doing so, f is always a function of M' .

3. Similarly, if ϕ is not deterministic but rather a noisy process that depends on exogenous randomness, one can include such randomness in the state (or history) of M (again this is redundant for M).

4. While the redundancy introduced in 2 and 3 keep most properties of M intact (including \mathcal{T} , π^* , Q^* , V^* , Q^π , V^π), it does affect the concentratability coefficient C defined in Assumption 5.

As an extreme example, consider an MDP whose state is always drawn i.i.d. from a fixed distribution independent of the time step or actions taken. Let μ be generated from taking actions uniformly at random, and ν generated by always taking the same action. In this case, $C = |\mathcal{A}|$ as the marginal of μ and ν on the states are exactly the same. However, when we treat histories (which include past actions) as states, C becomes exponentially large in horizon. So in this sense, Assumption 5 is also boundary-dependent. In contrast, Assumption 7 is invariant to such a transformation.

As a related side comment, our main text focuses on how the agent processes the sensory information, but there is another place where the agent interfaces with the environment—actions. While we do not discuss the boundary dependence of actions, we simply note that our boundary-invariant analyses are likely immune to any possible issues.

C Realizability is Not Verifiable

To show that realizability is not verifiable in general, it suffices to show an example in contextual bandits (Section 4). We further simplify the problem by restricting the number of actions to 1, which becomes a standard regression problem, and \mathcal{F} is realizable if it contains the Bayes-optimal predictor. We provide an argument below, inspired by that of the No Free Lunch theorem [Wolpert, 1996].

Consider a regression problem with finite feature space \mathcal{X} and label space $\mathcal{Y} = [0, 1]$. The hypothesis class only consists of one function, $f_{1/2}$, that takes a constant value 1/2. We will construct multiple data distributions in the form of $P_{X,Y} \in \Delta(\mathcal{X} \times \mathcal{Y})$, and $f_{1/2}$ is the Bayes-optimal regressor for one of them (hence realizable) but not for the others, and in the latter case realizability will be violated by

a large margin. An adversary chooses the distribution in a randomized manner, and the learner draws a finite dataset $\{(X_i, Y_i)\}$ from the chosen distribution and needs to decide whether $\mathcal{F} = \{f_{1/2}\}$ is realizable or not. We show that no learner can answer this question better than random guess when $|\mathcal{X}|$ goes to infinity.

In all distributions, the marginal of X is always uniform, and it remains to specify $P_{Y|X}$. For the realizable case, $P_{Y|X}$ is distributed as a Bernoulli random variable independent of the value of X . It will be convenient to refer to a data distribution by its Bayes-optimal regressor, so this distribution is labeled $f_{1/2}$.

For the remaining distributions, the label Y is always a deterministic and binary function of X , and there are in total $2^{|\mathcal{X}|}$ such functions. When the adversary chooses a distribution from this family, it always draws uniformly randomly, and we refer to the drawn function (and distribution) f_{rnd} . Note that regardless of which function is drawn, $f_{1/2}$ always violates realizability by a constantly large margin:

$$\mathbb{E}_X[(f_{1/2}(X) - f_{\text{rnd}}(X))^2] = 1/4.$$

The adversary chooses $f_{1/2}$ with $1/2$ probability, and f_{rnd} with $1/2$ probability. Since the learner only receives a finite sample, as long as there is no collision in $\{X_i\}$, there is no way to distinguish between $f_{1/2}$ and f_{rnd} . This is because, $\{(X_i, Y_i)\}$ can be drawn in two steps, where we first draw all the $\{X_i\}$ i.i.d. from $\text{Unif}(\mathcal{X})$, and this step does not reveal any information about the identity of the distribution. The second step generates $\{Y_i\}$ conditioned on $\{X_i\}$. Assuming no collision in $\{X_i\}$, it is easy to verify that the joint distribution over $\{Y_i\}$ is i.i.d. Bernoulli for both $f_{1/2}$ and f_{rnd} . Furthermore, fixing the sample size, the collision probability goes to 0 as $|\mathcal{X}|$ increases, and the learner cannot do better than a random guess.

Note that this hardness result does not apply when the learner has access to the resetting operations discussed in Section 3, as the learner can draw multiple Y 's from the same X to verify if $P_{Y|X}$ is stochastic ($f_{1/2}$) or deterministic (f_{rnd}) and succeed with high probability.

D Necessity of the Squared-loss Decomposition Condition

Here we provide an example showing the necessity of Eq.(5) in Assumption 4. In particular, if Eq.(5) is completely removed, the algorithm may fail to learn a valid value function in the limit of infinite data even when \mathcal{F} contains one.

Consider a simple contextual bandit problem with two contexts (states), s_A and s_B , and $d_0(s_A) = d_0(s_B) = 1/2$. The problem is uncontrolled (i.e., there is only one action and one policy), and both states yield deterministic reward 0.5. Let $\mathcal{F} = \{f_1, f_2\}$, where $f_1(s_A) = 1$, $f_1(s_B) = 0$, and $f_2(s_A) = f_2(s_B) = 0.6$. By Assumption 4 (with Eq.(5) removed), f_1 is a valid reward function while f_2 is not. However, $\mathcal{L}_\mu(f_1) = 0.25$ and $\mathcal{L}_\mu(f_2) = 0.01$, and the regression algorithm will pick f_2 with accurate estimation of the losses.

Further Comments In the above example, we note that there is nothing wrong in calling f_1 a valid reward function (though it is counter-intuitive). In fact, if this bandit problem is a part of a larger MDP—say it appears at the end of an episodic task, and d_0 is the only possible distribution that can be induced over s_A and s_B , then f_1 may well be part of an optimal value function, and a near-optimal policy can be learned via active exploration using the OLIVE algorithm [Jiang et al., 2017].¹⁵ The reason that f_1 should not be considered as a valid reward function in the context of Section 4 is due to the batch learning setting and the squared-loss regression algorithm. So Eq.(5) is a condition that is specific to the setting and the algorithm, and not inherent in our boundary-invariant definition of reward/value-functions.

¹⁵Formally, f_1 does not violate the validity condition in their Definition 3.

E Proofs of Sections 4 and 5

Proof of Theorem 2. $\mathbb{E}_{d_0 \times \pi_{f^*}}[f^*] - v^{\pi_f}$

$$\begin{aligned} &\leq \mathbb{E}_{s \sim d_0}[f^*(s, \pi_{f^*}) - \hat{f}(s, \pi_{f^*}) + \hat{f}(s, \pi_{\hat{f}}) - f^*(s, \pi_{\hat{f}}) + f^*(s, \pi_{\hat{f}})] - \mathbb{E}_{(s,a) \sim d_0 \times \pi_{\hat{f}}}[r] \\ &= \mathbb{E}_{d_0 \times \pi_{f^*}}[f^* - \hat{f}] + \mathbb{E}_{d_0 \times \pi_{\hat{f}}}[\hat{f} - f^*] + \mathbb{E}_{(s,a) \sim d_0 \times \pi_{\hat{f}}}[f^*(s, a) - r]. \end{aligned}$$

The 3rd term is 0 given that f^* is a valid reward function (Eq.(4)). Let ν be a placeholder for either $d_0 \times \pi_{\hat{f}}$ or $d_0 \times \pi_{f^*}$. For either of the first 2 terms, its square can be bounded as

$$\begin{aligned} &(\mathbb{E}_{\nu}[f^* - \hat{f}])^2 \leq \mathbb{E}_{\nu}[(f^* - \hat{f})^2] = \|f^* - \hat{f}\|_{\nu}^2 && \text{(Jensen)} \\ &\leq C \cdot \|f^* - \hat{f}\|_{\mu}^2 && \text{(Assumption 3)} \\ &= C \cdot (\mathcal{L}_{\mu}(\hat{f}) - \mathcal{L}_{\mu}(f^*)) && \text{(Eq.(5))} \\ &= C \cdot (\mathcal{L}_{\mu}(\hat{f}) - \min_{f \in \mathcal{F}} \mathcal{L}_{\mu}(f)) \leq C\epsilon. && \square \end{aligned}$$

Proof of Lemma 4. Let $\mathcal{B}_{r,s'}f$ be a shorthand for $r + \gamma \max_{a' \in \mathcal{A}} f(s', a')$. Also recall that $\mathbb{E}_{\nu}[\cdot]$ is short for $\mathbb{E}_{(s,a) \sim \nu, r \sim R(s,a), s' \sim P(s,a)}[\cdot]$, and $\mathbb{E}_{\nu}[f]$ for $\mathbb{E}_{\nu}[f(s, a)]$. The first step is to show that

$$\|\mathcal{B}f - \mathcal{B}f'\|_{\nu}^2 \leq \mathbb{E}_{\nu}[(\mathcal{B}_{r,s'}f - \mathcal{B}_{r,s'}f')^2]. \quad (9)$$

To prove this, we start with Eq.(7):

$$\begin{aligned} 0 &= \mathcal{L}_{\mu}(\mathcal{B}f; f') - \mathcal{L}_{\mu}(\mathcal{B}f'; f') - \|\mathcal{B}f - \mathcal{B}f'\|_{\mu}^2 \\ &= \mathbb{E}_{\nu}[(\mathcal{B}f - \mathcal{B}_{r,s'}f')^2] - \mathbb{E}_{\nu}[(\mathcal{B}f' - \mathcal{B}_{r,s'}f')^2] - \mathbb{E}_{\nu}[(\mathcal{B}f - \mathcal{B}f')^2] \\ &= 2 \mathbb{E}_{\nu}[(\mathcal{B}f' - \mathcal{B}_{r,s'}f')(\mathcal{B}f - \mathcal{B}f')]. \end{aligned}$$

Now we have

$$\mathbb{E}_{\nu}[(\mathcal{B}f - \mathcal{B}f')(\mathcal{B}f' - \mathcal{B}_{r,s'}f')] = 0, \quad (10)$$

and by symmetry

$$\mathbb{E}_{\nu}[(\mathcal{B}f' - \mathcal{B}f)(\mathcal{B}f - \mathcal{B}_{r,s'}f)] = 0. \quad (11)$$

We are ready to prove Eq.(9): its RHS is

$$\begin{aligned} &\mathbb{E}_{\nu}[(\mathcal{B}_{r,s'}f - \mathcal{B}_{r,s'}f')^2] \\ &= \mathbb{E}_{\nu}[(\mathcal{B}_{r,s'}f - \mathcal{B}_{r,s'}f' - \mathcal{B}f + \mathcal{B}f' + \mathcal{B}f - \mathcal{B}f')^2] \\ &= \mathbb{E}_{\nu}[(\mathcal{B}_{r,s'}f - \mathcal{B}_{r,s'}f' - \mathcal{B}f + \mathcal{B}f')^2] + \mathbb{E}_{\nu}[(\mathcal{B}f - \mathcal{B}f')^2] \\ &\quad + 2\mathbb{E}_{\nu}[(\mathcal{B}f - \mathcal{B}f')(\mathcal{B}_{r,s'}f - \mathcal{B}_{r,s'}f')] + 2\mathbb{E}_{\nu}[(\mathcal{B}f - \mathcal{B}f')(\mathcal{B}f' - \mathcal{B}_{r,s'}f')]. \end{aligned}$$

The 1st term is non-negative, the 2nd term is the LHS of Eq.(9), and the rest two terms are 0 according to Eq.(10) and (11). So Eq.(9) holds.

Now from the RHS of Eq.(9):

$$\begin{aligned} &\mathbb{E}_{\nu}[(\mathcal{B}_{r,s'}f - \mathcal{B}_{r,s'}f')^2] \\ &= \gamma^2 \mathbb{E}_{(s,a) \sim \nu, s' \sim P(s,a)}[(f(s', \pi_f) - f'(s', \pi_{f'}))^2] \\ &= \gamma^2 \mathbb{E}_{s' \sim P(\nu)}[(f(s', \pi_f) - f'(s', \pi_{f'}))^2] \\ &\leq \gamma^2 \mathbb{E}_{s' \sim P(\nu)}[(f(s', \pi_{f,f'}) - f'(s', \pi_{f,f'}))^2] \\ &= \gamma^2 \|f - f'\|_{P(\nu) \times \pi_{f,f'}}^2. \end{aligned} \quad \square$$

Proof of Theorem 3. Since $\mathcal{B}f \in \mathcal{F}$ for any $f \in \mathcal{F}$ by our definition, we can apply \mathcal{B} repeatedly to a function. Indeed, pick any $f \in \mathcal{F}$, we show that for large enough k , $\|\mathcal{B}^{k+1}f - \mathcal{B}^k f\|_{\nu} = 0$ for any admissible ν , so $\mathcal{B}^k f$ will satisfy the definition of f^* . This is because

$$\begin{aligned} \|\mathcal{B}^{k+1}f - \mathcal{B}^k f\|_{\nu} &\leq \gamma \|\mathcal{B}^k f - \mathcal{B}^{k-1}f\|_{P(\nu) \times \pi_{\mathcal{B}^k f, \mathcal{B}^{k-1}f'}} && \text{(Lemma 4)} \\ &\leq \gamma^2 \|\mathcal{B}^{k-1}f - \mathcal{B}^{k-2}f\|_{P(P(\nu) \times \pi_{\mathcal{B}^k f, \mathcal{B}^{k-1}f'}) \times \pi_{\mathcal{B}^{k-1}f, \mathcal{B}^{k-2}f'}} \\ &\leq \dots \leq \gamma^k \|\mathcal{B}f - f\|_{\square} \leq \gamma^k \|\mathcal{B}f - f\|_{\infty}, \end{aligned}$$

where \square is some admissible distribution. (Its detailed form is not important, but the reader can infer from the derivation above.) Given the boundedness of \mathcal{F} , $\|\mathcal{B}^{k+1}f - \mathcal{B}^k f\|_\nu$ becomes arbitrarily close to 0 for all ν uniformly as k increases. Now for each $f \in \mathcal{F}$, define $\mathcal{E}(f) := \sup_\nu \|\mathcal{B}f - f\|_\nu$. Since \mathcal{F} is finite, there exists a minimum non-zero value for $\mathcal{E}(f)$, so with large enough k , $\sup_\nu \|\mathcal{B}^{k+1}f - \mathcal{B}^k f\|_\nu$ will be smaller than such a minimum value and must be 0. \square

Comment In the proof of Theorem 3 we used the fact \mathcal{F} is finite to show that $\|\mathcal{B}^{k+1}f - \mathcal{B}^k f\| = 0$. This is the only place in this paper where we need the finiteness of \mathcal{F} . Even if \mathcal{F} is continuous, we can still use a large enough k to upper bound $\|\mathcal{B}^{k+1}f - \mathcal{B}^k f\|$ with an arbitrarily small number, which reduces the elegance of the theorem statements and has no impact on our results otherwise.

Proof of Theorem 5. We first show that f^* is a value-function of π_{f^*} on any admissible ν . The easiest way to prove this is to introduce the classical (boundary-dependent) notion of $Q^{\pi_{f^*}}$ as a bridge. Note that $\forall \nu$ we always have

$$\mathbb{E}_\nu[Q^{\pi_{f^*}}] = \mathbb{E}[\sum_{t=1}^{\infty} \gamma^{t-1} r_t \mid (s_1, a_1) \sim \nu, a_{2:\infty} \sim \pi_{f^*}].$$

So it suffices to show that $\forall \nu, \mathbb{E}_\nu[Q^{\pi_{f^*}}] = \mathbb{E}_\nu[f^*]$. We prove this using (a slight variant of) the value difference decomposition lemma [Jiang et al., 2017, Lemma 1]:

$$\mathbb{E}_\nu[f^*] - \mathbb{E}_\nu[Q^{\pi_{f^*}}] = \sum_{t=1}^{\infty} \gamma^{t-1} \mathbb{E}_{(s,a) \sim d_{\nu,t}^\pi} [f^*(s, a) - r - \gamma \max_{a' \in \mathcal{A}} f^*(s', a')].$$

Here with a slight abuse of notation we use $d_{\nu,t}^\pi$ to denote the distribution over (s_t, a_t) induced by $(s_1, a_1) \sim \nu, a_{2:t-1} \sim \pi$, and $\pi = \pi_{f^*}$. For each term on the RHS,

$$\begin{aligned} & \mathbb{E}_{(s,a) \sim d_{\nu,t}^\pi} [f^*(s, a) - r - \gamma \max_{a' \in \mathcal{A}} f^*(s', a')] \\ &= \mathbb{E}_{(s,a) \sim d_{\nu,t}^\pi} [f^*(s, a) - (\mathcal{B}f^*)(s, a) + (\mathcal{B}f^*)(s, a) - r - \gamma \max_{a' \in \mathcal{A}} f^*(s', a')] \\ &= \mathbb{E}_{d_{\nu,t}^\pi} [f^* - \mathcal{B}f^*] + \mathbb{E}_{(s,a) \sim d_{\nu,t}^\pi} [(\mathcal{B}f^*)(s, a) - r - \gamma \max_{a' \in \mathcal{A}} f^*(s', a')]. \end{aligned}$$

The second term is 0 because by the definition of \mathcal{B} (Assumption 8): $\mathcal{B}f^*$ is a reward function for random reward $r + \gamma \max_{a' \in \mathcal{A}} f^*(s', a')$ under any admissible distribution, including $d_{\nu,t}^\pi$. The first term is also 0 because

$$|\mathbb{E}_{d_{\nu,t}^\pi} [f^* - \mathcal{B}f^*]|^2 \leq \mathbb{E}_{d_{\nu,t}^\pi} [(f^* - \mathcal{B}f^*)^2] = 0. \quad (\text{Theorem 3})$$

Now

$$\begin{aligned} v^{\pi_{f^*}} - v^{\hat{\pi}} &= \sum_{t=1}^{\infty} \gamma^{t-1} \mathbb{E}_{(s,a) \sim d_t^{\hat{\pi}}} [Q^{\pi_{f^*}}(s, \pi_{f^*}) - Q^{\pi_{f^*}}(s, a)] \\ &\quad (\text{see e.g., Kakade and Langford [2002, Lemma 6.1]}) \\ &= \sum_{t=1}^{\infty} \gamma^{t-1} \mathbb{E}_{(s,a) \sim d_t^{\hat{\pi}}} [f^*(s, \pi_{f^*}) - f_k(s, \pi_{f^*}) + f_k(s, a) - f^*(s, a)] \\ &\quad (\mathbb{E}_\nu[Q^{\pi_{f^*}}] = \mathbb{E}_\nu[f^*]) \\ &\leq \sum_{t=1}^{\infty} \gamma^{t-1} \left(\|f_k - f^*\|_{\eta_t^{\hat{\pi}} \times \pi_{f^*}} + \|f_k - f^*\|_{d_t^{\hat{\pi}}} \right), \end{aligned} \quad (12)$$

where $\eta_t^{\hat{\pi}}$ is the marginal of $d_t^{\hat{\pi}}$ on states. Since both $d_t^{\hat{\pi}}$ and $\eta_t^{\hat{\pi}} \times \pi_{f^*}$ are admissible distributions, it suffices to upper-bound $\|f^* - f_k\|$ on any admissible distribution ν . In particular,

$$\begin{aligned} \|f_k - f^*\|_\nu &= \|f_k - \mathcal{B}f_{k-1} + \mathcal{B}f_{k-1} - f^*\|_\nu \\ &\leq \|f_k - \mathcal{B}f_{k-1}\|_\nu + \|\mathcal{B}f_{k-1} - \mathcal{B}f^*\|_\nu \\ &\leq \|f_k - \mathcal{B}f_{k-1}\|_\nu + \gamma \|f_{k-1} - f^*\|_{P(\nu) \times \pi_{f_{k-1}, f^*}}. \end{aligned} \quad (\text{Lemma 4})$$

Note that the second term is also w.r.t. an admissible distribution, so the inequalities can be expanded all the way to $\|f_0 - f^*\|$. For the first term,

$$\begin{aligned} \|f_k - \mathcal{B}f_{k-1}\|_\nu &\leq \sqrt{C} \|f_k - \mathcal{B}f_{k-1}\|_\mu \\ &= \sqrt{C} \sqrt{\mathcal{L}_\mu(f_k; f_{k-1}) - \mathcal{L}_\mu(\mathcal{B}f_{k-1}; f_{k-1})} \leq \sqrt{C}\epsilon. \end{aligned} \quad (\text{Eq.(5)})$$

Altogether we have on any admissible ν ,

$$\|f_k - f^*\|_\nu \leq \frac{\sqrt{C\epsilon}}{1-\gamma} + \gamma^k V_{\max}.$$

The proof is completed by applying this bound to Eq.(12). \square

F Boundary 0

There is an extreme choice of the boundary for every RL problem, where the environment part always has *deterministic* transition dynamics and a possibly stochastic initial state distribution. The construction has been given by Ng and Jordan [2000], and we briefly describe the idea here: All random transitions can be viewed as a deterministic transition function that takes an additional input, that is, there always exists a deterministic function T , such that

$$s \sim P(s, a) \Leftrightarrow s = T(s, a, \sigma),$$

where σ is a random variable from some suitable distribution (e.g., $\text{Unif}([0, 1])$). Now we augment the state representation of the MDP to include all the σ 's that we ever need to use in an episode, and generate them at the beginning of an episode (hence random initial state) so that all later transitions become deterministic. Of course, any realistic agent should not be able to observe the σ 's, and this restriction is reflected by the fact that any $f \in \mathcal{F}$ cannot depend on σ . If the environment is simulated on a computer, then “boundary 0” in Figure 1 is a good approximation of this situation, where the Pseudo-Random Generator (PRG) plays the role of σ 's.

Below we discuss a few topics in the context of this construction.

Algorithms for deterministic environments Many difficulties in RL arise due to stochastic transitions, and there are algorithms for deterministic environments that avoid these difficulties. For example, learning with bootstrapped target (e.g., temporal difference, Q -learning) can diverge under function approximation [Van Roy, 1994, Gordon, 1995, Tsitsiklis and Van Roy, 1997], but if the environment is deterministic, one can optimize $\mathbb{E}[(f(s, a) - r - \gamma \max_{a' \in \mathcal{A}} f(s', a'))^2]$ under an exploratory distribution to learn the Q^* , and the process is always convergent and enjoys superior theoretical properties.¹⁶ Now we just argued that all RL problems can be viewed as deterministic; why isn't everyone using the above algorithm instead of TD/ Q -learning?

The reason is that the algorithm tries to learn the Q^* of the deterministic environment defined by boundary 0, essentially competing with an omnipotent agent that has precise knowledge of the outcomes of all the random events ahead of time. Since the actual function approximator does not have access to such information, realizability will be severely impaired.

On the role of d_0 In Section 2 we specify an initial state distribution d_0 for the MDP. While this is common in modelling episodic tasks, a reasonable question to ask is what if the agent can start with any state (distribution) of its own choice. Our theory can actually handle this case pretty easily: we simply need to add all possible initial distributions and the downstream distributions induced by them into the definition of admissible distributions (Definition 2).

Without this modification, the theory will break down, and an obvious counterexample comes from the boundary 0 construction: If the agent is allowed to start from an arbitrary state deterministically, there will be no randomness in the trajectory, and the data sampled from such an initial state does not truthfully reflect the stochastic dynamics of the environment.

Should μ also be admissible? From the counterexample above, we see that non-admissible distributions can be problematic. This leads to the following question: Shouldn't μ be admissible, since it describes the data on which we run the learning algorithm? Interestingly, in Section 5 we did not make such an assumption and the analysis still went through. We do not have an intuitive answer as to why, and the only explanation is that Assumption 7 prevented the degenerate scenarios from happening.

¹⁶This is a special case of the residual algorithms introduced by Baird [1995]. The residual algorithms require double sampling (sampling two i.i.d. s' from the same (s, a)), which is not needed in deterministic environments.

G π^* Can be Useless in Imitation Learning

Here we show an example where π^* (for a poorly chosen boundary) is useless for the purpose of imitation learning. Consider any episodic RL problem that has no intermediate rewards and the terminal reward r is Bernoulli distributed and the mean lies in $[0.5, 1]$. We transform the problem in a way indistinguishable for the learner as follows: Whenever a random reward $r \sim \text{Bernoulli}(p)$ is given at the end, we replace that with a random transition to two states, each of which has two actions: in state s_A , action a_A yields a deterministic reward of +1 and a_B yields 0, and in state s_B , action a_B yields +1 and a_A yields 0. When the random reward in the original problem has mean $p \in [0.5, 1]$, the transition distribution over s_A and s_B in the transformed problem is p and $1 - p$, respectively. The identity information of s_A and s_B is not available to the agent (e.g., the function approximator treats the two states equivalently), so the transformed problem is completely equivalent to the original problem, except that the agent should always take a_A in s_A or s_B .¹⁷

Suppose that we generate demonstration data from π^* for the transformed problem and use an imitation learning algorithm to train an agent. Since π^* can distinguish between s_A and s_B , it will take a_A or a_B depending on the observed identity and always achieve a terminal reward of 1. In the original problem, the agent is in general supposed to take actions to maximize the value of p , but π^* in the transformed problem has no incentive to do so and can take arbitrary actions before reaching s_A or s_B , making the demonstrations useless to the bounded agent.

¹⁷While the transformed problem is partially observable to the learner, giving the history information to the learner’s state representation still does not help it leverage the demonstration, and we can remove partial observability by having multiple pairs of s_A and s_B , each of which can be only visited with a fixed Bernoulli distribution.