Analysis of a Classification-based Policy Iteration Algorithm

Alessandro Lazaric, Mohammad Ghavamzadeh, Remi Munos

Analysis of Classification-based Policy Iteration Algorithms

Alessandro Lazaric alessandro.lazaric@inria.fr
Mohammad Ghavamzadeh mohammad.ghavamzadeh@inria.fr
Rémi Munos remi.munos@inria.fr
INRIA Lille - Nord Europe, Team SequeL, France

Editor:

Abstract

We introduce a variant of the classification-based approach to policy iteration which uses a cost-sensitive loss function weighting each classification mistake by its actual *regret*, i.e., the difference between the action-value of the greedy action and of the action chosen by the classifier. For this algorithm, we provide a full finite-sample analysis. Our results state a performance bound in terms of the number of policy improvement steps, the number of rollouts used in each iteration, the capacity of the considered policy space (classifier), and a capacity measure which indicates how well the policy space can approximate policies that are greedy w.r.t. any of its members. The analysis reveals a tradeoff between the estimation and approximation errors in this classification-based policy iteration setting. Furthermore it confirms the intuition that classification-based policy iteration algorithms could be favorably compared to value-based approaches when the policies can be approximated more easily than their corresponding value functions. We also study the consistency of the algorithm when there exists a sequence of policy spaces with increasing capacity.

Keywords: reinforcement learning, policy iteration, classification-based approach to policy iteration, finite-sample analysis.

1. Introduction

Policy iteration (Howard, 1960) is a method of computing an optimal policy for any given Markov decision process (MDP). It is an iterative procedure that discovers a deterministic optimal policy by generating a sequence of monotonically improving policies. Each iteration k of this algorithm consists of two phases: *policy evaluation* in which the action-value function Q^{π_k} of the current policy π_k is computed, and *policy improvement* in which the new (improved) policy π_{k+1} is generated as the greedy policy w.r.t. Q^{π_k}, i.e., $\pi_{k+1}(x) = \arg\max_{a \in \mathcal{A}} Q^{\pi_k}(x,a)$. Unfortunately, in MDPs with large (or continuous) state and action spaces, the policy evaluation problem cannot be solved exactly and approximation techniques are required. In approximate policy iteration (API), a function approximation scheme is usually employed in the policy evaluation phase. The most common approach is to find a good approximation of the value function of π_k in a real-valued function space (see e.g., Bradtke and Barto 1996; Lagoudakis and Parr 2003a). The main drawbacks of this approach are: **1)** the action-value function, Q^{π_k}, is not known in advance and its high quality samples are often very expensive to obtain, if this option is possible at all, **2)** it is often difficult to find a function space rich enough to represent the action-value function accurately, and thus, careful hand-tuning is needed to achieve satisfactory results, **3)**

for the success of policy iteration, it is not necessary to estimate Q^{π_k} accurately at every state-action pair, what is important is to have an approximation of the action-value function whose greedy policy has a performance similar to the greedy policy w.r.t. the actual action-value function, and **4)** this method may not be the right choice in domains where good policies are easier to represent and learn than the corresponding value functions.

To address the above issues, mainly **3** and **4**,[1] variants of API have been proposed that replace the usual value function learning step (approximating the action-value function over the entire state-action space) with a learning step in a policy space (Lagoudakis and Parr, 2003b; Fern et al., 2004). The main idea is to cast the policy improvement step as a *classification* problem. The training set is generated using rollout estimates of Q^π over a finite number of states $\mathcal{D} = \{x_i\}_{i=1}^N$, called the *rollout set*, and for any action $a \in \mathcal{A}$.[2] For each $x \in \mathcal{D}$, if the estimated value $\widehat{Q}^\pi(x, a^*)$ is greater than the estimated value of all other actions with *high confidence*, the state-action pair (x, a^*) is added to the training set with a positive label. In this case, (x, a) for the rest of the actions are labeled negative and added to the training set. The policy improvement step thus reduces to solving a classification problem to find a policy in a given hypothesis space that best predicts the greedy action at every state. Although whether selecting a suitable policy space is any easier than a value function space is highly debatable, we can argue that the classification-based API methods can be advantageous in problems where good policies are easier to represent and learn than their value functions.

The classification-based API algorithms can be viewed as a type of reduction from reinforcement learning (RL) to classification, i.e., solving a MDP by generating and solving a series of classification problems. There have been other proposals for reducing RL to classification. Bagnell et al. (2003) introduced an algorithm for learning non-stationary policies in RL. For a specified horizon h, their approach learns a sequence of h policies. At each iteration, all policies are fixed except for one, which is optimized by forming a classification problem via policy rollout. Langford and Zadrozny (2005) provided a formal reduction from RL to classification, showing that ϵ-accurate classification implies near optimal RL. This approach uses an optimistic variant of sparse sampling to generate h classification problems, one for each horizon time step. The main limitation of this work is that it does not provide a practical method for generating training examples for these classification problems.

Although the classification-based API algorithms have been successfully applied to benchmark problems (Lagoudakis and Parr, 2003b; Fern et al., 2004) and have been modified to become more computationally efficient (Dimitrakakis and Lagoudakis, 2008b), a full theoretical understanding of them is still lacking. Fern et al. (2006) and Dimitrakakis and Lagoudakis (2008a) provide a preliminary theoretical analysis of their algorithm. In particular, they both bound the difference in performance at each iteration between the learned policy and the true greedy policy. Their analysis is limited to one step policy update (they do not show how the error in the policy update is propagated through the iterations of the API algorithm) and either to finite class of policies (in Fern et al., 2006) or to a specific architecture (a uniform grid in Dimitrakakis and Lagoudakis, 2008a). Moreover, the bound reported in Fern et al. (2006) depends inversely on the minimum Q-value gap between a

1. The first drawback is shared by all reinforcement learning algorithms and the second one is common to all practical applications of machine learning methods.
2. It is worth stressing that Q^π is estimated just on states in $\mathcal{D}$ and not over the entire state-action space.

greedy and a sub-greedy action over the state space. In some classes of MDPs this gap can be arbitrarily small so that the learned policy can be arbitrarily worse than the greedy policy. In order to deal with this problem Dimitrakakis and Lagoudakis (2008a) assume the action-value functions to be smooth and the probability of states with a small Q-value gap to be small.

In this paper, we derive a full finite-sample analysis of a classification-based API algorithm, called *direct policy iteration* (DPI). It is based on a cost-sensitive loss function weighting each classification error by its actual *regret*, i.e., the difference between the action-value of the greedy action and of the action chosen by DPI. Using this loss, we are able to derive a performance bound with no dependency on the minimum Q-value gap and no assumption on the probability of states with small Q-value gap. Our analysis further extends those in Fern et al. (2006) and Dimitrakakis and Lagoudakis (2008a) by considering arbitrary policy spaces, and by showing how the error at each step is propagated through the iterations of the API algorithm. We also analyze the consistency of DPI when there exists a sequence of policy spaces with increasing capacity. We first use a counterexample and show that DPI is not consistent in general, and then prove its consistency for the class of Lipschitz MDPs. We conclude the paper with a discussion on different theoretical and practical aspects of DPI.

The rest of the paper is organized as follows. In Section 2, we define the basic concepts and set up the notation used in the paper. Section 3 introduces the general classification-based approach to policy iteration and details the DPI algorithm. In Section 4, we provide a finite-sample analysis for the DPI algorithm. The approximation error and the consistency of the algorithm are discussed in Section 5. While all the main results are derived in case of two actions, i.e., $|\mathcal{A}| = 2$, in Section 6 we show how they can be extended to the general case of multiple actions. In Section 7, we conclude the paper and discuss the obtained results.

2. Preliminaries

In this section, we set the notation used throughout the paper. A discounted Markov decision process (MDP) $\mathcal{M}$ is a tuple $\langle \mathcal{X}, \mathcal{A}, r, p, \gamma \rangle$, where the state space $\mathcal{X}$ is a bounded closed subset of a Euclidean space $\mathbb{R}^d$, the set of actions $\mathcal{A}$ is finite ($|\mathcal{A}| < \infty$), the reward function $r : \mathcal{X} \times \mathcal{A} \to \mathbb{R}$ is uniformly bounded by $R_{\max}$, the transition model $p(\cdot|x, a)$ is a distribution over $\mathcal{X}$, and $\gamma \in (0, 1)$ is a discount factor. Let $\mathcal{B}^V(\mathcal{X}; V_{\max})$ and $\mathcal{B}^Q(\mathcal{X} \times \mathcal{A}; Q_{\max})$ be the space of Borel measurable value and action-value functions bounded by $V_{\max}$ and $Q_{\max}$ ($V_{\max} = Q_{\max} = \frac{R_{\max}}{1-\gamma}$), respectively. We also use $\mathcal{B}^\pi(\mathcal{X})$ to denote the space of deterministic policies $\pi : \mathcal{X} \to \mathcal{A}$. The value function of a policy π, V^π, is the unique fixed-point of the Bellman operator $\mathcal{T}^\pi : \mathcal{B}^V(\mathcal{X}; V_{\max}) \to \mathcal{B}^V(\mathcal{X}; V_{\max})$ defined by

$$(\mathcal{T}^\pi V)(x) = r\big(x, \pi(x)\big) + \gamma \int_{\mathcal{X}} p\big(dy|x, \pi(x)\big) V(y).$$

The action-value function Q^π is defined as

$$Q^\pi(x, a) = r(x, a) + \gamma \int_{\mathcal{X}} p(dy|x, a) V^\pi(y).$$

Similarly, the optimal value function, V^*, is the unique fixed-point of the optimal Bellman operator $\mathcal{T} : \mathcal{B}^V(\mathcal{X}; V_{\max}) \to \mathcal{B}^V(\mathcal{X}; V_{\max})$ defined as

$$(\mathcal{T}V)(x) = \max_{a \in \mathcal{A}} \left[r(x,a) + \gamma \int_{\mathcal{X}} p(dy|x,a) V(y) \right],$$

and the optimal action-value function Q^* is defined by

$$Q^*(x,a) = r(x,a) + \gamma \int_{\mathcal{X}} p(dy|x,a) V^*(y).$$

We say that a deterministic policy $\pi \in \mathcal{B}^\pi(\mathcal{X})$ is *greedy* w.r.t. an action-value function Q, if $\pi(x) \in \arg\max_{a \in \mathcal{A}} Q(x,a), \forall x \in \mathcal{X}$. Greedy policies are important because any greedy policy w.r.t. Q^* is optimal. We define the greedy policy operator $\mathcal{G} : \mathcal{B}^\pi(\mathcal{X}) \to \mathcal{B}^\pi(\mathcal{X})$ as[3]

$$(\mathcal{G}\pi)(x) = \arg\max_{a \in \mathcal{A}} Q^\pi(x,a). \tag{1}$$

In the analysis of this paper, $\mathcal{G}$ plays a role similar to the one played by the optimal Bellman operator, $\mathcal{T}$, in the analysis of the fitted value iteration algorithm (Munos and Szepesvári 2008, Section 5).

3. The DPI Algorithm

In this section, we outline the direct policy iteration (DPI) algorithm. DPI shares the same structure as the algorithms in Lagoudakis and Parr (2003b) and Fern et al. (2004). Although it can benefit from improvements in **1)** selecting states for the rollout set $\mathcal{D}$, **2)** the criteria used to add a sample to the training set, and **3)** the rollout strategy, as discussed in Lagoudakis and Parr (2003b) and Dimitrakakis and Lagoudakis (2008b), here we consider its basic form in order to ease the analysis.

In DPI, at each iteration k, a new policy π_{k+1} is computed from π_k, as the best approximation of $\mathcal{G}\pi_k$, by solving a cost-sensitive classification problem. More formally, DPI is based on the following loss function.

Definition 1 *The loss function at iteration k for a policy π is denoted by $\ell_{\pi_k}(\cdot;\pi)$ and is defined as*

$$\ell_{\pi_k}(x;\pi) = \max_{a \in \mathcal{A}} Q^{\pi_k}(x,a) - Q^{\pi_k}\big(x, \pi(x)\big), \qquad \forall x \in \mathcal{X}.$$

Given a distribution ρ over $\mathcal{X}$, we define the expected error as the expectation of the loss function $\ell_{\pi_k}(\cdot;\pi)$ according to ρ, [4]

$$\mathcal{L}_{\pi_k}(\rho;\pi) = \int_{\mathcal{X}} \ell_{\pi_k}(x;\pi)\rho(dx) = \int_{\mathcal{X}} \Big[\max_{a \in \mathcal{A}} Q^{\pi_k}(x,a) - Q^{\pi_k}\big(x, \pi(x)\big) \Big] \rho(dx). \tag{2}$$

3. In Equation 1, the tie among the actions maximizing $Q^\pi(x,a)$ is broken in an arbitrary but consistent manner.
4. The expected error $\mathcal{L}_{\pi_k}(\rho;\pi)$ can be seen as the $L_{1,\rho}$-norm of the loss function.

Input: policy space $\Pi \subseteq \mathcal{B}^\pi(\mathcal{X})$, state distribution ρ, number of rollout states N, number of rollouts per state-action pair M
Initialize: Let $\pi_0 \in \Pi$ be an arbitrary policy
for $k = 0, 1, 2, \ldots$ **do**
  Construct the rollout set $\mathcal{D}_k = \{x_i\}_{i=1}^N$, $x_i \overset{\text{iid}}{\sim} \rho$
  for all states $x_i \in \mathcal{D}_k$ and actions $a \in \mathcal{A}$ **do**
    for $j = 1$ to M **do**
      Perform a rollout according to policy π_k and return $R_j^{\pi_k}(x_i, a) = r(x_i, a) + \sum_{t=1}^{H-1} \gamma^t r(x^t, \pi_k(x^t))$, $x^t \sim p\big(\cdot\,|x^{t-1}, \pi_k(x^{t-1})\big)$ and $x^1 \sim p(\cdot|x_i, a)$
    end for
    $\widehat{Q}^{\pi_k}(x_i, a) = \frac{1}{M} \sum_{j=1}^M R_j^{\pi_k}(x_i, a)$
  end for
  $\pi_{k+1} = \arg\min_{\pi \in \Pi} \widehat{\mathcal{L}}_{\pi_k}(\widehat{\rho}; \pi)$ **(classifier)**
end for

Figure 1: The Direct Policy Iteration (DPI) algorithm.

While in Lagoudakis and Parr (2003b) the goal is to minimize the number of misclassifications, i.e., they use a 0/1 loss function, DPI learns a policy which aims at minimizing the error $\mathcal{L}_{\pi_k}$. Similar to other classification-based RL algorithms (Fern et al., 2004; Langford and Zadrozny, 2005; Li et al., 2007), DPI does not focus on finding a uniformly accurate approximation of the actions taken by the greedy policy, but rather on finding actions leading to a similar performance. This is consistent with the final objective of policy iteration, which is to obtain a policy with similar performance to an optimal policy, and not necessarily one that takes actions similar to an optimal policy.[5]

As illustrated in Figure 1, for each state $x_i \in \mathcal{D}_k$ and for each action $a \in \mathcal{A}$, an estimate of the action-value function of the current policy is computed through M independent rollouts. A H-horizon rollout of a policy π_k for a state-action pair (x_i, a) is

$$R^{\pi_k}(x_i, a) = r(x_i, a) + \sum_{t=1}^{H-1} \gamma^t r\big(x^t, \pi_k(x^t)\big), \tag{3}$$

where $x^t \sim p\big(\cdot\,|x^{t-1}, \pi_k(x^{t-1})\big)$ and $x^1 \sim p(\cdot|x_i, a)$. The action-value function estimation is then obtained by averaging M independent rollouts $\{R_j^{\pi_k}(x_i, a)\}_{1 \le j \le M}$ as

$$\widehat{Q}^{\pi_k}(x_i, a) = \frac{1}{M} \sum_{j=1}^M R_j^{\pi_k}(x_i, a). \tag{4}$$

Given the outcome of the rollouts, the empirical loss is defined as follows.

Definition 2 *For any $x \in \mathcal{D}_k$, the empirical loss function at iteration k for a policy π is*

$$\widehat{\ell}_{\pi_k}(x; \pi) = \max_{a \in \mathcal{A}} \widehat{Q}^{\pi_k}(x, a) - \widehat{Q}^{\pi_k}\big(x, \pi(x)\big),$$

5. We refer the readers to Li et al. (2007) for a simple example in which a good approximation (in terms of the number of mismatch in selecting actions) of the greedy policy has a very poor performance w.r.t. it.

where $\widehat{Q}^{\pi_k}(x,a)$ is a H-horizon rollout estimation of the action-value of π_k in (x,a) as defined by Equations 3 and 4. Similar to Definition 1, the empirical error is defined as the average over states in $\mathcal{D}_k$ of the empirical loss, [6]

$$\widehat{\mathcal{L}}_{\pi_k}(\widehat{\rho};\pi) = \frac{1}{N}\sum_{i=1}^{N}\Big[\max_{a\in\mathcal{A}}\widehat{Q}^{\pi_k}(x_i,a) - \widehat{Q}^{\pi_k}\big(x_i,\pi(x_i)\big)\Big],$$

where $\widehat{\rho}$ is the empirical distribution induced by the samples in $\mathcal{D}_k$.

Finally, DPI makes use of a classifier which returns a policy that minimizes the empirical error $\widehat{\mathcal{L}}_{\pi_k}(\widehat{\rho};\pi)$ over the policy space Π.

4. Finite-sample Analysis of DPI

In this section, we first provide a finite-sample analysis of the error incurred at each iteration of DPI in Theorem 5, and then show how this error is propagated through the iterations of the algorithm in Theorem 7. In the analysis, we explicitly assume that the action space contains only two actions, i.e., $\mathcal{A} = \{a_1, a_2\}$ and $|\mathcal{A}| = 2$. We will discuss this assumption and other theoretical and practical aspects of DPI in Section 6.

4.1 Error Bound at Each Iteration

Here we study the error incurred at each iteration k of the DPI algorithm. As it can be noticed by comparing the definition of the expected and empirical error, there are two sources of error in the algorithm of Figure 1. The first one depends on the use of a finite number of samples N in the rollout set to approximate the expectation w.r.t. the distribution ρ. The following lemma shows that the difference between the approximation obtained by averaging over the samples in the rollout set and the true expectation can be controlled and reduces to zero as the number of states grows.

Lemma 3 *Let Π be a policy space with finite VC-dimension $h = VC(\Pi) < \infty$ and $N > 0$ be the number of states in the rollout set $\mathcal{D}_k$, drawn i.i.d. from the state distribution ρ, then*

$$\mathbb{P}_{\mathcal{D}_k}\left[\sup_{\pi\in\Pi}\Big|\mathcal{L}_{\pi_k}(\widehat{\rho};\pi) - \mathcal{L}_{\pi_k}(\rho;\pi)\Big| > \epsilon\right] \le \delta\ ,$$

with $\epsilon = 16 Q_{\max}\sqrt{\frac{2}{N}\left(h\log\frac{eN}{h} + \log\frac{8}{\delta}\right)}$.

Proof Let $\mathcal{F}_k$ be the space of the loss functions at iteration k induced by the policies in Π, i.e., $\mathcal{F}_k = \{\ell_{\pi_k}(\cdot;\pi)|\ \pi\in\Pi\}$. Note that all the functions $\ell_{\pi_k}(\cdot;\pi)\in\mathcal{F}_k$ are uniformly bounded by $2Q_{\max}$. By Pollard's inequality (Pollard, 1984), for the bounded space $\mathcal{F}_k$, we have

$$\mathbb{P}_{\mathcal{D}_k}\left[\sup_{\ell_{\pi_k}\in\mathcal{F}_k}\Big|\frac{1}{N}\sum_{i=1}^{N}\ell_{\pi_k}(x_i) - \int\ell_{\pi_k}(x)\rho(dx)\Big| > \epsilon\right] \le 8\mathbb{E}\left[\mathcal{N}_1\left(\frac{\epsilon}{8},\mathcal{F}_k,X_1^N\right)\right]\exp\left(-\frac{N\epsilon^2}{128(2Q_{\max})^2}\right).$$

6. Alternatively, the empirical error can be seen as the $L_{1,\widehat{\rho}}$-norm of the empirical loss.

Note that at each iteration k, the policy π_k is a random variable because it is the minimizer of the empirical error $\widehat{\mathcal{L}}_{\pi_{k-1}}(\widehat{\rho};\pi)$. However, π_k depends only on the previous policies and rollout sets up to $\mathcal{D}_{k-1}$, and is completely independent of the samples in $\mathcal{D}_k$, thus Pollard's inequality applies. We now show how the covering number of the space $\mathcal{F}_k$ can be directly related to the VC-dimension of Π. First we rewrite the loss function as $\ell_{\pi_k}(x;\pi) = \mathbb{I}\{(\mathcal{G}\pi_k)(x) \neq \pi(x)\}\,\Delta^{\pi_k}(x)$, where

$$\Delta^{\pi_k}(x) = \max_{a\in\mathcal{A}} Q^{\pi_k}(x,a) - \min_{a'\in\mathcal{A}} Q^{\pi_k}(x,a') \tag{5}$$

is the gap between the two actions (i.e., the regret of choosing the wrong action). Let $\bar{\Pi}$ be an $\frac{\epsilon}{2Q_{\max}}$-cover of Π using the empirical distance defined by the number of different actions at the states $\{x_i\}_{1\le i\le N}$, then $\bar{\mathcal{F}}_k = \{\bar{\ell}_{\pi_k}(\cdot) = \ell_{\pi_k}(\cdot;\bar{\pi})|\bar{\pi}\in\bar{\Pi}\}$ is an ϵ-cover of $\mathcal{F}_k$. In fact for any $\ell_{\pi_k}\in\mathcal{F}_k$, there exist a $\bar{\ell}_{\pi_k}\in\bar{\mathcal{F}}_k$ such that

$$\begin{aligned}
\frac{1}{N}\sum_{i=1}^{N}\left|\ell_{\pi_k}(x_i) - \bar{\ell}_{\pi_k}(x_i)\right| &= \frac{1}{N}\sum_{i=1}^{N}\Big|\mathbb{I}\{(\mathcal{G}\pi_k)(x_i)\neq\pi(x_i)\}\,\Delta^{\pi_k}(x_i) \\
&\quad -\mathbb{I}\{(\mathcal{G}\pi_k)(x_i)\neq\bar{\pi}(x_i)\}\,\Delta^{\pi_k}(x_i)\Big| \\
&\le 2Q_{\max}\frac{1}{N}\sum_{i=1}^{N}\left|\mathbb{I}\{(\mathcal{G}\pi_k)(x_i)\neq\pi(x_i)\} - \mathbb{I}\{(\mathcal{G}\pi_k)(x_i)\neq\bar{\pi}(x_i)\}\right| \\
&= 2Q_{\max}\frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\{(\pi(x_i)\neq\bar{\pi}(x_i)\} \le 2Q_{\max}\frac{\epsilon}{2Q_{\max}} = \epsilon.
\end{aligned}$$

Thus, we can now relate the covering number of $\mathcal{F}_k$ to the VC-dimension of Π

$$\mathcal{N}_1\left(\frac{\epsilon}{8},\mathcal{F}_k,X_1^N\right) \le \mathcal{N}_1\left(\frac{\epsilon}{16Q_{\max}},\Pi,X_1^N\right) \le S_\Pi(N) \le \left(\frac{eN}{h}\right)^h,$$

where $S_\Pi(N)$ is the growth function of Π and the last inequality follows from Sauer's lemma. Since $\mathcal{L}_{\pi_k}(\widehat{\rho};\pi) = \frac{1}{N}\sum_{i=1}^{N}\ell_{\pi_k}(x_i;\pi)$ and $\mathcal{L}_{\pi_k}(\rho;\pi) = \int \ell_{\pi_k}(x;\pi)\rho(dx)$, the final statement is obtained by inverting the Pollard's bound. ■

The other source of approximation in the algorithm of Figure 1 is due to the use of rollout estimates of the action-value function on the states in the rollout set. We define the true action-value for a state-action pair (x,a) with a finite horizon H as

$$Q_H^{\pi_k}(x,a) = \mathbb{E}\left[r(x,a) + \sum_{t=1}^{H-1}\gamma^t r\big(x^t,\pi_k(x^t)\big)\right].$$

It is easy to see that the H-horizon rollout estimates are stochastic estimations of $Q_H^{\pi_k}(x,a)$ which in turn satisfy

$$|Q^{\pi_k}(x,a) - Q_H^{\pi_k}(x,a)| = \left|\mathbb{E}\left[\sum_{t=H}^{\infty}\gamma^t r\big(x^t,\pi_k(x^t)\big)\right]\right| \le \gamma^H Q_{\max}. \tag{6}$$

We are now ready to prove the main result of this section. We show a high probability bound on the expected error at each iteration k of DPI.

In the proof of the main theorem we also need to bound the difference between the action values estimated with rollouts and the true action values. We thus report the following lemma.

Lemma 4 *Let Π be a policy space with finite VC-dimension $h = VC(\Pi) < \infty$ and $x_1, \dots, x_N$ be an arbitrary sequence of states. In each state we simulate M independent truncated rollouts, then*

$$\mathbb{P}\left[\sup_{\pi\in\Pi}\Big|\frac{1}{N}\sum_{i=1}^{N}\frac{1}{M}\sum_{j=1}^{M}R_j^{\pi_k}(x_i,\pi(x_i)) - \frac{1}{N}\sum_{i=1}^{N}Q_H^{\pi_k}(x_i,\pi(x_i))\Big| > \epsilon\right] \le \delta\ ,$$

with $\epsilon = 8(1-\gamma^H)Q_{\max}\sqrt{\frac{2}{MN}\left(h\log\frac{eMN}{h} + \log\frac{8}{\delta}\right)}$.

Proof Similar to the proof of Lemma 3, we rely on the Pollard's inequality to prove the statement. We first introduce a sequence of random events ω_{ij} such that for any $i = 1, \dots, N$ the event ω_{ij} is independently drawn from a suitable distribution ν_i. As a result, we may rewrite the rollout random variables as $R_j^{\pi_k}\big(x_i,\pi(x_i)\big) = R^{\pi_k}(\omega_{ij};\pi)$ and the statement of the theorem as

$$\mathbb{P}\left[\sup_{\pi\in\Pi}\Big|\frac{1}{MN}\sum_{i,j}R^{\pi_k}(\omega_{ij};\pi) - \frac{1}{MN}\sum_{i,j}\mathbb{E}_{\nu_i}\big[R^{\pi_k}(\omega_{ij};\pi)\big]\Big| > \epsilon\right] \le \delta.$$

Let $\mathcal{H}_k$ be the space of the rollout functions induced by the policies in Π at iteration k, i.e., $\mathcal{H}_k = \{R^{\pi_k}(\cdot;\pi)|\ \pi\in\Pi\}$. Note that all the functions $R^{\pi_k}(\cdot;\pi) \in \mathcal{H}_k$ are uniformly bounded by $(1-\gamma^H)Q_{\max}$. By Pollard's inequality (Pollard, 1984), for the bounded space $\mathcal{H}_k$, we have [7]

$$\mathbb{P}\left[\sup_{\pi\in\Pi}\Big|\frac{1}{MN}\sum_{i,j}R^{\pi_k}(\omega_{ij};\pi) - \frac{1}{MN}\sum_{i,j}\mathbb{E}_{\nu_i}[R^{\pi_k}(\omega_{ij};\pi)]\Big| > \epsilon\right]$$
$$\le 8\mathbb{E}\left[\mathcal{N}_1\left(\frac{\epsilon}{8},\mathcal{H}_k,\omega_1^{MN}\right)\right]\exp\left(-\frac{MN\epsilon^2}{128(1-\gamma^H)^2Q_{\max}^2}\right).$$

We now show how the covering number of the space $\mathcal{H}_k$ is related to the VC-dimension of Π. Let $\bar\Pi$ be an $\frac{\epsilon}{2(1-\gamma^H)Q_{\max}}$-cover of Π using the empirical distance defined at the states $\{x_i\}_{1\le i\le N}$, then $\bar{\mathcal{H}}_k = \{\bar R^{\pi_k}(\cdot) = R^{\pi_k}(\cdot;\bar\pi)|\bar\pi\in\bar\Pi\}$ is an ϵ-cover of $\mathcal{H}_k$. In fact for any $R^{\pi_k}\in\mathcal{H}_k$, there exist a $\bar R^{\pi_k}\in\bar{\mathcal{H}}_k$ such that

$$\frac{1}{MN}\sum_{i,j}\big|R^{\pi_k}(\omega_{ij}) - \bar R^{\pi_k}(\omega_{ij})\big| = \frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}\big|R_j^{\pi_k}\big(x_i,\pi(x_i)\big) - R_j^{\pi_k}\big(x_i,\bar\pi(x_i)\big)\big|$$
$$\le 2(1-\gamma^H)Q_{\max}\frac{1}{N}\sum_{i=1}^{N}\mathbb{I}\{\pi(x_i)\neq\bar\pi(x_i)\} \le 2(1-\gamma^H)Q_{\max}\frac{\epsilon}{2(1-\gamma^H)Q_{\max}} = \epsilon.$$

7. Note that since here the samples are independent but not identically distributed, we use a slight variation of the standard Pollard's inequality. We refer the reader to the proof of Pollard's inequality (e.g., Pollard 1984 or Devroye et al. 1996) to see that the standard proof can be easily extended to this case.

We can now relate the covering number of $\mathcal{F}_k$ to the VC-dimension of Π

$$\mathcal{N}_1\left(\frac{\epsilon}{8}, \mathcal{F}_k, \omega_1^{MN}\right) \leq \mathcal{N}_1\left(\frac{\epsilon}{16(1-\gamma^H)Q_{\max}}, \Pi, \omega_1^{MN}\right) \leq S_\Pi(MN) \leq \left(\frac{eMN}{h}\right)^h,$$

where $S_\Pi(N)$ is the growth function of Π and the last inequality follows from Sauer's lemma. The final statement is obtained by inverting the Pollard's bound. ■

Theorem 5 *Let Π be a policy space with finite VC-dimension $h = VC(\Pi) < \infty$ and ρ be a distribution over the state space $\mathcal{X}$. Let N be the number of states in $\mathcal{D}_k$ drawn i.i.d. from ρ at each iteration, H be the horizon of the rollouts, and M be the number of rollouts per state-action pair used in the estimation of the action-value functions. Let $\pi_{k+1} = \arg\min_{\pi\in\Pi} \widehat{\mathcal{L}}_{\pi_k}(\widehat{\rho}; \pi)$ be the policy computed at the k'th iteration of DPI . Then, for any $\delta > 0$, we have*

$$\mathcal{L}_{\pi_k}(\rho; \pi_{k+1}) \leq \inf_{\pi\in\Pi} \mathcal{L}_{\pi_k}(\rho; \pi) + 2(\epsilon_1 + \epsilon_2 + \gamma^H Q_{\max}), \tag{7}$$

with probability $1-\delta$, where

$$\epsilon_1 = 16Q_{\max}\sqrt{\frac{2}{N}\left(h\log\frac{eN}{h} + \log\frac{32}{\delta}\right)} \quad \text{and} \quad \epsilon_2 = 8(1-\gamma^H)Q_{\max}\sqrt{\frac{2}{MN}\left(h\log\frac{eMN}{h} + \log\frac{32}{\delta}\right)}.$$

Remark 1 The bound in Equation 7 can be decomposed into an approximation error $\inf_{\pi\in\Pi} \mathcal{L}_{\pi_k}(\rho; \pi)$ and an estimation error consisting of three terms ϵ_1, ϵ_2, and $\gamma^H Q_{\max}$. This is similar to generalization bounds in classification, where the approximation error is the distance between the target function (here the greedy policy w.r.t. π_k) and the function space Π. The first estimation term, ϵ_1, grows with the capacity of Π, measured by its VC-dimension h, and decreases with the number of sampled states N. Thus in order to avoid overfitting, we should have $N \gg h$. The second estimation term, ϵ_2, comes from the error in the estimation of the action-values due to the finite number of rollouts M. It is important to note the nice rate of $1/\sqrt{MN}$ instead of $1/\sqrt{M}$. This is due to the fact that we do not need a uniformly good estimation of the action-value function at all sampled states, but only an averaged estimation of those values at the sampled points. An important consequence of this is that the algorithm works perfectly well if we consider only $M = 1$ rollout per state-action. Therefore, given a fixed budget (number of rollouts per iteration) and a fixed rollout horizon H, the best allocation of M and N would be to choose $M = 1$ and sample as many states as possible, thus, reducing the risk of overfitting. The third estimation term, $\gamma^H Q_{\max}$, is due to the fact that we consider a finite horizon H for the rollouts. This term decreases as the rollout horizon H grows.

Remark 2 In Remark 1, we considered the tradeoff between the number of states, N, and the number of rollouts at each state-action pair, M, when a finite budget (number of rollouts per iteration) is given. It is also interesting to analyze the tradeoff with the rollout horizon, H, when the number of interactions with the generative model is fixed to

a maximum value $S = N \times M \times H$. The term γ^H decreases exponentially with a rate depending on γ, thus, it easy to see that by setting $M = 1$, a rough optimization of the bound in Theorem 5 leads to $H = O(\frac{\log S}{\log 1/\gamma})$ and $N = O(S/H)$. Similar to the tradeoff between M and N, this suggests that most of the resources should be allocated so as to have a large number of states, while the rollouts may have a fairly short horizon. Nonetheless, it is clear from the value of H that the discount factor is critical, and when it approaches 1 the horizon increases correspondingly.

Proof Let $a^*(x) = \arg\max_{a \in \mathcal{A}} Q^{\pi_k}(x, a)$ be the greedy action in state x.[8] We prove the following series of inequalities:

$$
\begin{aligned}
\mathcal{L}_{\pi_k}(\rho; \pi_{k+1}) &\overset{(a)}{\leq} \mathcal{L}_{\pi_k}(\widehat{\rho}; \pi_{k+1}) + \epsilon_1 && \text{w.p. } 1-\delta' \\
&= \frac{1}{N} \sum_{i=1}^{N} \Big[Q^{\pi_k}(x_i, a^*) - Q^{\pi_k}\big(x_i, \pi_{k+1}(x_i)\big) \Big] + \epsilon_1 \\
&\overset{(b)}{\leq} \frac{1}{N} \sum_{i=1}^{N} \Big[Q^{\pi_k}(x_i, a^*) - Q_H^{\pi_k}\big(x_i, \pi_{k+1}(x_i)\big) \Big] + \epsilon_1 + \gamma^H Q_{\max} && \text{w.p. } 1-\delta' \\
&\overset{(c)}{\leq} \frac{1}{N} \sum_{i=1}^{N} \Big[Q^{\pi_k}(x_i, a^*) - \widehat{Q}^{\pi_k}\big(x_i, \pi_{k+1}(x_i)\big) \Big] + \epsilon_1 + \epsilon_2 + \gamma^H Q_{\max} && \text{w.p. } 1-2\delta' \\
&\overset{(d)}{\leq} \frac{1}{N} \sum_{i=1}^{N} \Big[Q^{\pi_k}(x_i, a^*) - \widehat{Q}^{\pi_k}\big(x_i, \pi^*(x_i)\big) \Big] + \epsilon_1 + \epsilon_2 + \gamma^H Q_{\max} \\
&\overset{(e)}{\leq} \frac{1}{N} \sum_{i=1}^{N} \Big[Q^{\pi_k}(x_i, a^*) - Q_H^{\pi_k}\big(x_i, \pi^*(x_i)\big) \Big] + \epsilon_1 + 2\epsilon_2 + \gamma^H Q_{\max} && \text{w.p. } 1-3\delta' \\
&\overset{(f)}{\leq} \frac{1}{N} \sum_{i=1}^{N} \Big[Q^{\pi_k}(x_i, a^*) - Q^{\pi_k}\big(x_i, \pi^*(x_i)\big) \Big] + \epsilon_1 + 2(\epsilon_2 + \gamma^H Q_{\max}) && \text{w.p. } 1-3\delta' \\
&= \mathcal{L}_{\pi_k}(\widehat{\rho}; \pi^*) + \epsilon_1 + 2(\epsilon_2 + \gamma^H Q_{\max}) \\
&\overset{(g)}{\leq} \mathcal{L}_{\pi_k}(\rho; \pi^*) + 2(\epsilon_1 + \epsilon_2 + \gamma^H Q_{\max}) && \text{w.p. } 1-4\delta' \\
&= \inf_{\pi' \in \Pi} \mathcal{L}_{\pi_k}(\rho; \pi') + 2(\epsilon_1 + \epsilon_2 + \gamma^H Q_{\max}).
\end{aligned}
$$

The statement of the theorem is obtained by $\delta' = \delta/4$.

(a) It is an immediate application of Lemma 3, bounding the difference between $\mathcal{L}_{\pi_k}(\rho; \pi)$ and $\mathcal{L}_{\pi_k}(\widehat{\rho}; \pi)$ for any policy $\pi \in \Pi$.
(b) We use the inequality in Equation 6.

8. To simplify the notation, we remove the dependency of a^* on states and use a^* instead of $a^*(x)$ in the following.

(c) Here we introduce the estimated action-value function $\widehat{Q}^{\pi_k}$ by bounding[9]

$$\max_{a\in\mathcal{A}}\left[\frac{1}{N}\sum_{i=1}^{N}\widehat{Q}^{\pi_k}(x_i,a)-\frac{1}{N}\sum_{i=1}^{N}Q_H^{\pi_k}(x_i,a)\right],$$

the maximum over actions of the difference between the true action-value function with horizon H and its rollout estimates averaged over the states in the rollout set $\mathcal{D}_k=\{x_i\}_{i=1}^N$. For a fixed action a, by using Chernoff-Hoeffding inequality and by recalling the definition of $\widehat{Q}^{\pi_k}\big(x_i,a\big)$ as the average of M rollouts, we obtain

$$\frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}R_j^{\pi_k}(x_i,a)-\frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}Q_H^{\pi_k}(x_i,a)\le(1-\gamma^H)Q_{\max}\sqrt{\frac{2}{MN}\log\frac{1}{\delta'}}\,,$$

with probability $1-\delta'$. Therefore, by taking the union bound over actions, we have

$$\max_{a\in\mathcal{A}}\left[\frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}R_j^{\pi_k}(x_i,a)-\frac{1}{MN}\sum_{i=1}^{N}\sum_{j=1}^{M}Q_H^{\pi_k}(x_i,a)\right]\le(1-\gamma^H)Q_{\max}\sqrt{\frac{2}{MN}\log\frac{|\mathcal{A}|}{\delta'}}\,,$$

with probability $1-\delta'$.

(d) From the definition of π_{k+1} in the DPI algorithm (see Figure 1), we have

$$\pi_{k+1}=\arg\min_{\pi\in\Pi}\widehat{\mathcal{L}}_{\pi_k}(\widehat{\rho};\pi)=\arg\max_{\pi\in\Pi}\frac{1}{N}\sum_{i=1}^{N}\widehat{Q}^{\pi_k}\big(x_i,\pi(x_i)\big),$$

thus, $-\frac{1}{N}\sum_{i=1}^{N}\widehat{Q}^{\pi_k}\big(x_i,\pi_{k+1}(x_i)\big)$ can be maximized by replacing π_{k+1} with any other policy, particularly with

$$\pi^*=\arg\inf_{\pi'\in\Pi}\int_{\mathcal{X}}\Big(\max_{a\in\mathcal{A}}Q^{\pi_k}(x,a)-Q^{\pi_k}\big(x,\pi'(x)\big)\Big)\rho(dx).$$

(e)-(g) The final result follows by using Definition 6 and by applying the Chernoff-Hoeffding inequality, the inequality of Equation 6, and the regression generalization bound. ■

4.2 Error Propagation

In this section, we first show how the expected error is propagated through the iterations of DPI. We then analyze the error between the value function of the policy obtained by DPI after K iterations and the optimal value function in μ-norm, where μ is a distribution used to assess the performance of the algorithm which might be different from the sampling distribution ρ.

Before stating the main result, we define the *inherent greedy error* of a policy space Π.

9. The maximum over actions appears because π_{k+1} is a random variable whose randomness depends on the rollout set $\mathcal{D}_k$ and the rollouts generated in the k'th iteration of the DPI algorithm.

Definition 6 *We define the inherent greedy error of a policy space* $\Pi \subseteq \mathcal{B}^\pi(\mathcal{X})$ *as*

$$d(\Pi, \mathcal{G}\Pi) = \sup_{\pi \in \Pi} \inf_{\pi' \in \Pi} \mathcal{L}_\pi(\rho; \pi').$$

In other words, the inherent greedy error is the worst expected error that a error-minimizing policy $\pi' \in \Pi$ can incur in approximating the greedy policy $\mathcal{G}\pi,\ \pi \in \Pi$. This measures how well Π is able to approximate policies that are greedy w.r.t. any policy in Π.

Let P^π be the transition kernel for policy π, i.e., $P^\pi(dy|x) = p\big(dy|x, \pi(x)\big)$. It defines two related operators: a right-linear operator, $P^\pi \cdot$, which maps any $V \in \mathcal{B}^V(\mathcal{X}; V_{\max})$ to $(P^\pi V)(x) = \int V(y) P^\pi(dy|x)$, and a left-linear operator, $\cdot P^\pi$, that returns $(\mu P^\pi)(dy) = \int P^\pi(dy|x)\mu(dx)$ for any distribution μ over $\mathcal{X}$.

From the definitions of ℓ_{π_k}, $\mathcal{T}^\pi$, and $\mathcal{T}$, we have $\ell_{\pi_k}(\pi_{k+1}) = \mathcal{T}V^{\pi_k} - \mathcal{T}^{\pi_{k+1}}V^{\pi_k}$. We deduce the following pointwise inequalities:

$$\begin{aligned} V^{\pi_k} - V^{\pi_{k+1}} &= \mathcal{T}^{\pi_k}V^{\pi_k} - \mathcal{T}^{\pi_{k+1}}V^{\pi_k} + \mathcal{T}^{\pi_{k+1}}V^{\pi_k} - \mathcal{T}^{\pi_{k+1}}V^{\pi_{k+1}} \\ &\le \ell_{\pi_k}(\pi_{k+1}) + \gamma P^{\pi_{k+1}}(V^{\pi_k} - V^{\pi_{k+1}}) \,, \end{aligned}$$

which gives us $V^{\pi_k} - V^{\pi_{k+1}} \le (I - \gamma P^{\pi_{k+1}})^{-1}\ell_{\pi_k}(\pi_{k+1})$. Since $\mathcal{T}V^{\pi_k} \ge \mathcal{T}^{\pi^*}V^{\pi_k}$, we also have

$$\begin{aligned} V^* - V^{\pi_{k+1}} &= \mathcal{T}V^* - \mathcal{T}V^{\pi_k} + \mathcal{T}V^{\pi_k} - \mathcal{T}^{\pi_{k+1}}V^{\pi_k} + \mathcal{T}^{\pi_{k+1}}V^{\pi_k} - \mathcal{T}^{\pi_{k+1}}V^{\pi_{k+1}} \\ &\le \gamma P^*(V^* - V^{\pi_k}) + \ell_{\pi_k}(\pi_{k+1}) + \gamma P^{\pi_{k+1}}(V^{\pi_k} - V^{\pi_{k+1}}) \,, \end{aligned}$$

which yields

$$\begin{aligned} V^* - V^{\pi_{k+1}} &\le \gamma P^*(V^* - V^{\pi_k}) + \big[\gamma P^{\pi_{k+1}}(I - \gamma P^{\pi_{k+1}})^{-1} + I\big]\ell_{\pi_k}(\pi_{k+1}) \\ &= \gamma P^*(V^* - V^{\pi_k}) + (I - \gamma P^{\pi_{k+1}})^{-1}\ell_{\pi_k}(\pi_{k+1}) \,. \end{aligned}$$

Finally, by defining the operator $E_k = (I - \gamma P^{\pi_{k+1}})^{-1}$, which is well defined since $P^{\pi_{k+1}}$ is a stochastic kernel and $\gamma < 1$, and by induction, we obtain

$$V^* - V^{\pi_K} \le (\gamma P^*)^K (V^* - V^{\pi_0}) + \sum_{k=0}^{K-1} (\gamma P^*)^{K-k-1} E_k \ell_{\pi_k}(\pi_{k+1}) \,. \tag{8}$$

Equation 8 shows how the error at each iteration k of DPI, $\ell_{\pi_k}(\pi_{k+1})$, is propagated through the iterations and appears in the final error of the algorithm: $V^* - V^{\pi_K}$. Since we are interested in bounding the final error in μ-norm, which might be different than the sampling distribution ρ, we use one of the following assumptions:

Assumption 1 *For any policy* π *and any non-negative integers* s *and* t*, there exists a constant* $C_{\mu,\rho}(s,t) < \infty$ *such that* $\mu(P^*)^s(P^\pi)^t \le C_{\mu,\rho}(s,t)\rho$*. We define* $C_{\mu,\rho} = (1-\gamma)^2 \sum_{s=0}^\infty \sum_{t=0}^\infty \gamma^{s+t} C_{\mu,\rho}(s,t)$*.*

Assumption 2 *For any* $x \in \mathcal{X}$ *and any* $a \in \mathcal{A}$*, there exist a constant* $C_\rho < \infty$ *such that* $p(\cdot|x,a) \le C_\rho \rho(\cdot)$*.*

Note that *concentrability coefficients* similar to $C_{\mu,\rho}$ and C_ρ were previously used in the L_p-analysis of fitted value iteration (Munos, 2007; Munos and Szepesvári, 2008) and approximate policy iteration (Antos et al., 2008). See also (Farahmand et al., 2010) for a refined analysis. We now state our main result.

Theorem 7 *Let Π be a policy space with finite VC-dimension h and π_K be the policy generated by DPI after K iterations. Let M be the number of rollouts per state-action and N be the number of samples drawn i.i.d. from a distribution ρ over $\mathcal{X}$ at each iteration of DPI. Then, for any $\delta > 0$, we have*

$$||V^* - V^{\pi_K}||_{1,\mu} \leq \frac{C_{\mu,\rho}}{(1-\gamma)^2}\Big[d(\Pi, \mathcal{G}\Pi) + 2(\epsilon_1 + \epsilon_2 + \gamma^H Q_{\max})\Big] + \frac{2\gamma^K R_{\max}}{1-\gamma}, \quad \textit{(Assumption 1)}$$

$$||V^* - V^{\pi_K}||_{\infty} \leq \frac{C_{\rho}}{(1-\gamma)^2}\Big[d(\Pi, \mathcal{G}\Pi) + 2(\epsilon_1 + \epsilon_2 + \gamma^H Q_{\max})\Big] + \frac{2\gamma^K R_{\max}}{1-\gamma}, \quad \textit{(Assumption 2)}$$

with probability $1-\delta$, where

$$\epsilon_1 = 16 Q_{\max}\sqrt{\frac{2}{N}\left(h\log\frac{eN}{h} + \log\frac{32K}{\delta}\right)} \quad \textit{and} \quad \epsilon_2 = 8(1-\gamma^H)Q_{\max}\sqrt{\frac{2}{MN}\left(h\log\frac{eMN}{h} + \log\frac{32K}{\delta}\right)}.$$

Proof We have $C_{\mu,\rho} \leq C_\rho$ for any μ. Thus, if the L_1-bound holds for any μ, choosing μ to be a Dirac at each state implies that the L_∞-bound holds as well. Hence, we only need to prove the L_1-bound. By taking the absolute value point-wise in Equation 8 we obtain

$$|V^* - V^{\pi_K}| \leq (\gamma P^*)^K |V^* - V^{\pi_0}| + \sum_{k=0}^{K-1} (\gamma P^*)^{K-k-1}(I - \gamma P^{\pi_{k+1}})^{-1}|\ell_{\pi_k}(\pi_{k+1})| \,.$$

From the fact that $|V^* - V^{\pi_0}| \leq \frac{2}{1-\gamma} R_{\max}\mathbf{1}$, and by integrating both sides w.r.t. μ, and using Assumption 1 we have

$$||V^* - V^{\pi_K}||_{1,\mu} \leq \frac{2\gamma^K}{1-\gamma} R_{\max} + \sum_{k=0}^{K-1}\sum_{t=0}^{\infty} \gamma^{K-k-1}\gamma^t C_{\mu,\rho}(K-k-1, t)\mathcal{L}_{\pi_k}(\rho; \pi_{k+1}) \,.$$

From the definition of $C_{\mu,\rho}$ we obtain

$$||V^* - V^{\pi_K}||_{1,\mu} \leq \frac{2\gamma^K}{1-\gamma} R_{\max} + \frac{C_{\mu,\rho}}{(1-\gamma)^2} \max_{0\leq k\leq K} \mathcal{L}_{\pi_k}(\rho; \pi_{k+1}) \,.$$

By bounding $\mathcal{L}_{\pi_k}(\rho; \pi_{k+1})$ using Theorem 5 with a union bound argument over the K iterations and the definition of the inherent greedy error the claim follows. ■

5. Approximation Error

In Section 4.2, we analyzed how the expected error at each iteration k of DPI, $\mathcal{L}_{\pi_k}(\rho, \pi_{k+1})$, propagates through iterations. The final approximation error term in Theorem 7 is the

inherent greedy error of Definition 6, $d(\Pi, \mathcal{G}\Pi)$, which depends on the MDP and the richness of the policy space Π. The main question in this section is whether this approximation error can be made small by increasing the capacity of the policy space Π. The answer is not obvious because when the space of policies, Π, grows, it can better approximate any greedy policy w.r.t. a policy in Π, however, the number of such greedy policies grows as well. We start our analysis of this approximation error by introducing the notion of *universal family of policy spaces*.

Definition 8 *A sequence of policy spaces $\{\Pi_n\}$ is a universal family of policy spaces, if there exists a sequence of real numbers $\{\beta_n\}$ with $\lim_{n\to\infty} \beta_n = 0$, such that for any $n > 0$, Π_n is induced by a partition $P_n = \{\mathcal{X}_i\}_{i=1}^{S_n}$ over the state space $\mathcal{X}$ (i.e., for each S_n-tuple $(b_1, \ldots, b_{S_n})$ with $b_i \in \{0, 1\}$, there exists a policy $\pi \in \Pi_n$ such that $\pi(x) = b_i$ for all $x \in \mathcal{X}_i$ and for all $i \in \{1, \ldots, S_n\}$) such that*

$$\max_{1\le i\le S_n} \max_{x,y\in\mathcal{X}_i} ||x - y|| \le \beta_n.$$

This definition requires that for any $n > 0$, Π_n be the space of policies induced by a partition P_n, and the diameters of the elements $\mathcal{X}_i$ of this partition shrink to zero as n goes to infinity. The main property of such a sequence of spaces is that any fixed policy π can be approximated arbitrary well by policies of Π_n when $n \to \infty$. Although other definitions of universality could be used, Definition 8 seems natural and it is satisfied by widely-used classifiers such as k-nearest neighbor, uniform grid, and histogram.

In the next section, we first show that the universality of a policy space (Definition 8) does not guarantee that $d(\Pi_n, \mathcal{G}\Pi_n)$ converges to zero in a general MDP. In particular, we present a MDP in which $d(\Pi_n, \mathcal{G}\Pi_n)$ is constant (does not depend on n) even when $\{\Pi_n\}$ is a universal family of classifiers. We then prove that in Lipschitz MDPs, $d(\Pi_n, \mathcal{G}\Pi_n)$ converges to zero for a universal family of policy spaces.

5.1 Counterexample

In this section, we illustrate a simple example in which $d(\Pi_n, \mathcal{G}\Pi_n)$ does not go to zero, even when $\{\Pi_n\}$ is a universal family of classifiers. We consider a MDP with state space $\mathcal{X} = [0, 1]$, action space $\mathcal{A} = \{0, 1\}$, and the following transitions and rewards

$$x_{t+1} = \begin{cases} \min(x_t + 0.5, 1) & \text{if } a = 1, \\ x_t & \text{otherwise,} \end{cases} \qquad r(x, a) = \begin{cases} 0 & \text{if } x = 1, \\ R_1 & \text{else if } a = 1, \\ R_0 & \text{otherwise,} \end{cases}$$

where $(1 - \gamma^2)R_1 < R_0 < R_1$. (9)

We consider the policy space Π_n of piecewise constant policies obtained by uniformly partitioning the state space $\mathcal{X}$ into n intervals. This family of policy spaces is universal. The inherent greedy error of Π_n, $d(\Pi_n, \mathcal{G}\Pi_n)$, can be decomposed into the sum of the expected errors at each interval

$$d(\Pi_n, \mathcal{G}\Pi_n) = \sup_{\pi\in\Pi_n} \inf_{\pi'\in\Pi_n} \sum_{i=1}^{n} \mathcal{L}_\pi^{(i)}(\rho; \pi') \ ,$$

where $\mathcal{L}_\pi^{(i)}(\rho;\pi')$ is the same as $\mathcal{L}_\pi(\rho;\pi')$, only the integral is over the i'th interval instead of the entire state space $\mathcal{X}$. In the following we show that for the MDP and the universal class of policies considered here, $d(\Pi_n, \mathcal{G}\Pi_n)$ does not converge to zero as n grows.

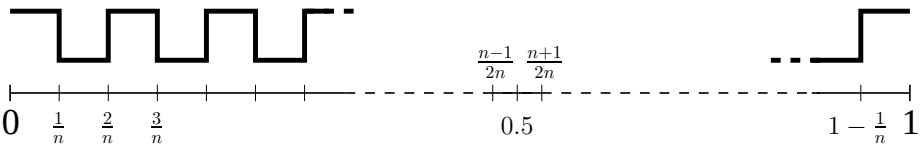

Figure 2: The policy used in the counterexample. It is one in odd and zero in even intervals. Note that the number of intervals, n, is assumed to be odd.

Let n be odd and $\pi \in \Pi_n$ be one in odd and zero in even intervals (see Figure 2). For any $x > 0.5$, the agent either stays in the same state forever by taking action 0, or goes out of bound in one step by taking action 1. Thus, given the assumption of Equation 9, it can be shown that for any x belonging to the intervals $i \geq \frac{n+1}{2}$ (the interval containing 0.5 and above), $(\mathcal{G}\pi)(x) = 0$. This means that there exists a policy $\pi' \in \Pi_n$ such that $\mathcal{L}_\pi^{(i)}(\rho;\pi') = 0$ for all the intervals $i \geq \frac{n+1}{2}$. However, $\mathcal{G}\pi$ does not remain constant in the intervals $i \leq \frac{n-1}{2}$, and changes its value in the middle of the interval. Using Equation 9, we can show that

$$\inf_{\pi' \in \Pi_n} \sum_{i=1}^{n} \mathcal{L}_\pi^{(i)}(\rho;\pi') = C\Big(1 + \frac{1}{1-\gamma}\Big)\frac{n-1}{8n} \geq \frac{C}{16}\Big(1 + \frac{1}{1-\gamma}\Big)\,,$$

where $C = \min\{(1-\gamma)(R_1 - R_0), R_0 - (1-\gamma^2)R_1\}$. This means that for any odd n, it is always possible to find a policy $\pi \in \Pi_n$ such that $\inf_{\pi' \in \Pi_n} \mathcal{L}_\pi(\rho;\pi')$ is lower bounded by a constant independent of n, thus $\lim_{n\to\infty} d(\Pi_n, \mathcal{G}\Pi_n) \neq 0$.

5.2 Lipschitz MDPs

In this section, we prove that for Lipschitz MDPs, $d(\Pi_n, \mathcal{G}\Pi_n)$ goes to zero when $\{\Pi_n\}$ is a universal family of classifiers. We start by defining a Lipschitz MDP.

Definition 9 *A MDP is Lipschitz if both its transition probability and reward functions are Lipschitz, i.e.,* $\forall (B, x, x', a) \in \mathcal{B}(\mathcal{X}) \times \mathcal{X} \times \mathcal{X} \times \mathcal{A}$

$$\begin{aligned} |r(x,a) - r(x',a)| &\leq L_r \|x - x'\|, \\ |p(B|x,a) - p(B|x',a)| &\leq L_p \|x - x'\|, \end{aligned}$$

with L_r and L_p being the Lipschitz constants of the transitions and reward, respectively.

An important property of Lipschitz MDPs is that for any function $Q \in \mathcal{B}^Q(\mathcal{X}\times\mathcal{A}; Q_{\max})$, the function obtained by applying the Bellman operator $\mathcal{T}^\pi$ to $Q(\cdot,a)$, $(\mathcal{T}^\pi Q)(\cdot,a)$, is Lipschitz with constant $L = (L_r + \gamma Q_{\max} L_p)$, for any action $a \in \mathcal{A}$. As a result, the function $Q^\pi(\cdot,a)$, which is the unique fixed point of the Bellman operator $\mathcal{T}^\pi$, is Lipschitz with constant L, for any policy $\pi \in \mathcal{B}^\pi(\mathcal{X})$ and any action $a \in \mathcal{A}$.

Theorem 10 *Let $\mathcal{M}$ be a Lipschitz MDP with $|\mathcal{A}| = 2$ and $\{\Pi_n\}$ be a universal family of policy spaces (Definition 8). Then $\lim_{n\to\infty} d(\Pi_n, \mathcal{G}\Pi_n) = 0$.*

Proof

$$
\begin{aligned}
d(\Pi_n, \mathcal{G}\Pi_n) &= \sup_{\pi\in\Pi_n} \inf_{\pi'\in\Pi_n} \int_{\mathcal{X}} \ell_\pi(x;\pi')\rho(dx) \\
&\overset{(a)}{=} \sup_{\pi\in\Pi_n} \inf_{\pi'\in\Pi_n} \int_{\mathcal{X}} \mathbb{I}\left\{(\mathcal{G}\pi)(x) \neq \pi'(x)\right\} \Delta^\pi(x)\rho(dx) \\
&\overset{(b)}{=} \sup_{\pi\in\Pi_n} \inf_{\pi'\in\Pi_n} \sum_{i=1}^{S_n} \int_{\mathcal{X}_i} \mathbb{I}\left\{(\mathcal{G}\pi)(x) \neq \pi'(x)\right\} \Delta^\pi(x)\rho(dx) \\
&\overset{(c)}{=} \sup_{\pi\in\Pi_n} \sum_{i=1}^{S_n} \min_{a\in\mathcal{A}} \int_{\mathcal{X}_i} \mathbb{I}\left\{(\mathcal{G}\pi)(x) \neq a\right\} \Delta^\pi(x)\rho(dx) \\
&\overset{(d)}{\leq} \sup_{\pi\in\Pi_n} \sum_{i=1}^{S_n} \min_{a\in\mathcal{A}} \int_{\mathcal{X}_i} \mathbb{I}\left\{(\mathcal{G}\pi)(x) \neq a\right\} 2L \inf_{y:\Delta^\pi(y)=0} \|x-y\| \, \rho(dx) \\
&\overset{(e)}{\leq} 2L \sup_{\pi\in\Pi_n} \sum_{i=1}^{S_n} \min_{a\in\mathcal{A}} \int_{\mathcal{X}_i} \mathbb{I}\left\{(\mathcal{G}\pi)(x) \neq a\right\} \beta_n \rho(dx) \\
&\overset{(f)}{\leq} 2L\beta_n \sum_{i=1}^{S_n} \int_{\mathcal{X}_i} \rho(dx) = 2L\beta_n.
\end{aligned}
$$

(a) We rewrite Definition 6, where Δ^π is the regret of choosing the wrong action defined by Equation 5.
(b) Since Π_n contains piecewise constants policies induced by the partition $P_n = \{\mathcal{X}_i\}$, we split the integral as the sum over the regions.
(c) Since the policies in Π_n can take any action in each possible region, the policy π' minimizing the loss is the one which takes the best action in each region.
(d) Since $\mathcal{M}$ is Lipschitz, both $\max_{a\in\mathcal{A}} Q^\pi(\cdot, a)$ and $\min_{a'\in\mathcal{A}} Q^\pi(\cdot, a')$ are Lipschitz, and thus, $\Delta^\pi(\cdot)$ is $2L$-Lipschitz. Furthermore, Δ^π is zero in all the states in which the policy $\mathcal{G}\pi$ changes (see Figure 3). Thus, for any state x the value $\Delta^\pi(x)$ can be bounded using the Lipschitz property by taking y as the closest state to x in which $\Delta^\pi(y) = 0$.
(e) If $\mathcal{G}\pi$ is constant in a region $\mathcal{X}_i$, the integral can be made zero by setting a to the greedy action (thus making $\mathbb{I}\{(\mathcal{G}\pi)(x) \neq a\} = 0$ for any $x \in \mathcal{X}_i$). Otherwise if $\mathcal{G}\pi$ changes in a state $y \in \mathcal{X}_i$, then $\Delta^\pi(y) = 0$ and we can replace $||x - y||$ by the diameter of the region which is bounded by β_n according to the definition of the universal family of spaces (Definition 8).
(f) We simply take $\mathbb{I}\{(\mathcal{G}\pi)(x) \neq a\} = 1$ in each region.
The claim follows using the definition of the universal family of policy spaces. ■

Theorem 10 together with the counter-example in Section 5.1 show that the assumption on the policy space is not enough to guarantee a small approximation error and additional assumptions on the smoothness of the MDP (e.g., Lipschitz condition) must be satisfied.

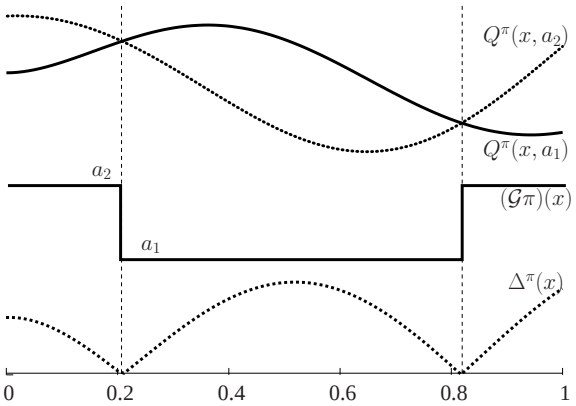

Figure 3: This figure is used as an illustrative example in the proof of Theorem 10. It shows the action-value function of a Lipschitz MDP for a policy π, $Q^\pi(\cdot, a_1)$ and $Q^\pi(\cdot, a_2)$ (top), the corresponding greedy policy $\mathcal{G}\pi$ (middle), and the regret of selecting the wrong action, Δ^π, (bottom).

5.3 Consistency of DPI

A highly desirable property of any learning algorithm is *consistency*, i.e., as the number of samples grows to infinity, the error of the algorithm converges to zero. It can be seen that as the number of samples N and the rollout horizon H grow in Theorem 5, ϵ_1 and ϵ_2 become arbitrarily small, and thus, the expected error at each iteration, $\mathcal{L}_{\pi_k}(\rho; \pi_{k+1})$, is bounded by the inherent greedy error $d(\Pi, \mathcal{G}\Pi)$. We can conclude from the results of this section that DPI is not consistent in general, but it is consistent for the class of Lipschitz MDPs, when a universal family of policy spaces is used. However, it is important to note that as we increase the index n also the capacity of the policy space Π (its VC-dimension h) grows as well, and thus, when the number of samples N goes to infinity, in order to still have a vanishing the estimation error (ϵ_1 in Theorem 5), we should guarantee that N grows faster than $VC(\Pi)$. We deduce the following result.

Corollary 11 *Let $\mathcal{M}$ be a Lipschitz MDP with $|\mathcal{A}| = 2$, $\{\Pi_n\}$ be a universal family of policy spaces (Definition 8), $h(n) = VC(\Pi_n)$, and $\lim_{n,N\to\infty} \frac{h(n)}{N} = 0$. Then DPI is consistent:*

$$\lim_{\substack{n,\, N,\, H,\, K \,\to\, \infty \\ \delta \,\to\, 0}} V^{\pi_K} = V^* \,, \qquad \text{w.p. } 1.$$

6. Extension to Multiple Actions

The analysis of Sections 4 and 5 are for the case that the action space contains only two actions. In Section 6.1 we extend the previous theoretical analysis to the general case of an action space with $|\mathcal{A}| > 2$. While the theoretical analysis is completely independent from the specific algorithm used to solve the empirical error minimization problem (see DPI algorithm of Figure 1), in Section 6.2 we discuss which algorithms could be employed to solve this problem in the case of multiple actions.

6.1 Theoretical Analysis

From the theoretical point of view, the extension of the previous results to multiple actions is straightforward. The definitions of loss and error functions do not change and we just need to use an alternative complexity measure for multi-class classification. We rely on the following definitions from Ben-David et al. (1995).

Definition 12 *Let $\Pi \subseteq \mathcal{B}^\pi(\mathcal{X})$ be a set of deterministic policies and $\Psi = \{\psi : \mathcal{A} \to \{0,1,*\}\}$ be a set of mappings from the action space to the set $\{0,1,*\}$. A finite set of N states $\mathcal{X}_N = \{x_i\}_{i=1}^N \subseteq \mathcal{X}$ is Ψ-shattered by Π if there exists a vector of mappings $\psi^N = \left(\psi^{(1)}, \dots, \psi^{(N)}\right)^\top \in \Psi^N$ such that for any vector $v \in \{0,1\}^N$, there exist a policy $\pi \in \Pi$ such that $\psi^{(i)} \circ \pi(x_i) = v_i, \quad 1 \le i \le N$. The Ψ-dimension of Π is the maximal cardinality of a subset of $\mathcal{X}$, Ψ-shattered by Π.*

Definition 13 *Let $\Pi \subseteq \mathcal{B}^\pi(\mathcal{X})$ be a set of deterministic policies and $\Psi = \{\psi_{k,l} : \mathcal{A} \to \{0,1,*\},\ 1 \le k \ne l \le L\}$ be a set of possible mappings such that*

$$\psi_{k,l}(a) = \begin{cases} 1 & \text{if } a = k, \\ 0 & \text{if } a = l, \\ * & \text{otherwise}, \end{cases}$$

then the Natarajan dimension of Π, N-dim(Π), is the Ψ-dimension of Π.

By using a policy space with finite Natarajan dimension, we derive the following corollary to Theorem 5.

Corollary 14 *Let $\Pi \subseteq \mathcal{B}^\pi(\mathcal{X})$ be a policy space with finite Natarajan dimension $h = $ N-dim(Π) $< \infty$. Let ρ be a distribution over the state space $\mathcal{X}$, N be the number of states in $\mathcal{D}_k$ drawn i.i.d. from ρ, and M be the number of rollouts per state-action pair used by DPI in the estimation of the action-value functions. Let $\pi_{k+1} = \arg\min_{\pi \in \Pi} \widehat{\mathcal{L}}_{\pi_k}(\widehat{\rho}; \pi)$ be the policy computed at the k'th iteration of DPI. Then, for any $\delta > 0$, we have*

$$\mathcal{L}_{\pi_k}(\rho; \pi_{k+1}) \le \inf_{\pi \in \Pi} \mathcal{L}_{\pi_k}(\rho; \pi) + 2(\epsilon_1 + \epsilon_2 + \gamma^H Q_{\max}), \tag{10}$$

with probability $1 - \delta$, where

$$\epsilon_1 = 16 Q_{\max} \sqrt{\frac{2}{N}\left(h \log \frac{|\mathcal{A}| e (N+1)^2}{h} + \log \frac{32}{\delta}\right)} \qquad \text{and} \qquad \epsilon_2 = (1-\gamma^H) Q_{\max} \sqrt{\frac{2}{MN} \log \frac{4|\mathcal{A}|}{\delta}}\ .$$

Proof In order to prove this corollary we just need a minor change in Lemma 3, which now becomes a concentration of measures inequality for a space of multi-class classifiers Π with finite Natarajan dimension. By using similar steps as in the proof of Lemma 3 and by recalling the Sauer's lemma for finite Natarajan dimension spaces (Ben-David et al., 1995), we obtain

$$\mathbb{P}\left[\sup_{\pi \in \Pi} \left| \mathcal{L}_{\pi_k}(\widehat{\rho}; \pi) - \mathcal{L}_{\pi_k}(\rho; \pi) \right| > \epsilon \right] \le \delta\ ,$$

with $\epsilon = 16 Q_{\max} \sqrt{\frac{2}{N}\left(h \log \frac{|\mathcal{A}|e(N+1)^2}{h} + \log \frac{8}{\delta}\right)}$. The rest of the proof is exactly the same as in Theorem 5. ■

Similarly, the consistency analysis in case of Lipschitz MDPs remains mostly unaffected by the introduction of multiple actions.

Corollary 15 *Let $\{\Pi_n\}$ be a universal family of policy spaces (Definition 8), and $\mathcal{M}$ be a Lipschitz MDP (Definition 9). Then* $\lim_{n\to\infty} d(\Pi_n, \mathcal{G}\Pi_n) = 0$.

Proof The critical part in the proof is the definition of the gap function, which now compares the performance of the greedy action to the performance of the action chosen by the policy π':

$$\Delta^{\pi,\pi'}(x) = \max_{a\in\mathcal{A}} Q^\pi(x,a) - Q^\pi\big(x, \pi'(x)\big).$$

Note that $\Delta^{\pi,\pi'}(\cdot)$ is no longer a Lipschitz function because it is a function of x through the policy π'. However, $\Delta^{\pi,\pi'}(x)$ is Lipschitz in each region $\mathcal{X}_i$, $i = 1\ldots, S_n$, because in each region $\mathcal{X}_i$, by the definition of the policy space, π' is forced to be constant. Therefore, in a region $\mathcal{X}_i$ in which $\pi'(x) = a$, $\forall x \in \mathcal{X}_i$, $\Delta^{\pi,\pi'}(x)$ may be written as

$$\Delta^{\pi,\pi'}(x) = \Delta^{\pi,a}(x) = \max_{a'\in\mathcal{A}} Q^\pi(x,a') - Q^\pi(x,a).$$

The proof here is exactly the same as in Theorem 10 up to step **(c)**, and then we have

$$\begin{aligned}
d(\Pi_n, \mathcal{G}\Pi_n) &= \sup_{\pi\in\Pi_n} \inf_{\pi'\in\Pi_n} \int_{\mathcal{X}} \ell_\pi(x;\pi')\rho(dx) \\
&= \sup_{\pi\in\Pi_n} \inf_{\pi'\in\Pi_n} \int_{\mathcal{X}} \mathbb{I}\left\{(\mathcal{G}\pi)(x) \neq \pi'(x)\right\} \Delta^{\pi,\pi'}(x)\rho(dx) \\
&= \sup_{\pi\in\Pi_n} \inf_{\pi'\in\Pi_n} \sum_{i=1}^{S_n} \int_{\mathcal{X}_i} \mathbb{I}\left\{(\mathcal{G}\pi)(x) \neq \pi'(x)\right\} \Delta^{\pi,\pi'}(x)\rho(dx) \\
&= \sup_{\pi\in\Pi_n} \sum_{i=1}^{S_n} \min_{a\in\mathcal{A}} \int_{\mathcal{X}_i} \mathbb{I}\left\{(\mathcal{G}\pi)(x) \neq a\right\} \Delta^{\pi,a}(x)\rho(dx) \\
&\leq \sup_{\pi\in\Pi_n} \sum_{i=1}^{S_n} \min_{a\in\mathcal{A}} \int_{\mathcal{X}_i} \Delta^{\pi,a}(x)\rho(dx). \qquad (11)
\end{aligned}$$

If the greedy action does not change in a region $\mathcal{X}_i$, i.e., $\forall x \in \mathcal{X}_i$, $(\mathcal{G}\pi)(x) = a'$, for an action $a' \in \mathcal{A}$, then the minimizing policy π' must select action a' in $\mathcal{X}_i$, and thus, the loss will be zero in $\mathcal{X}_i$. Now let assume that the greedy action changes at a state $y \in \mathcal{X}_i$ and the action $b_i \in \arg\max_{a\in\mathcal{A}} Q^\pi(y,a)$. In this case, we have

$$\min_{a\in\mathcal{A}} \int_{\mathcal{X}_i} \Delta^{\pi,a}(x)\rho(dx) \leq \int_{\mathcal{X}_i} \Delta^{\pi,b_i}(x)\rho(dx) \leq \int_{\mathcal{X}_i} \big(\Delta^{\pi,b_i}(y) + 2L\|x-y\|\big)\rho(dx),$$

since the function $x \mapsto \Delta^{\pi,b_i}(x)$ is $2L$-Lipschitz. Now since $\Delta^{\pi,b_i}(y) = 0$, we deduce from Equation 11 that

$$d(\Pi_n, \mathcal{G}\Pi_n) \leq \sup_{\pi \in \Pi_n} \sum_{i=1}^{S_n} \int_{\mathcal{X}_i} 2L||x-y||\rho(dx) \leq \sup_{\pi \in \Pi_n} \sum_{i=1}^{S_n} \int_{\mathcal{X}_i} 2L\beta_n \rho(dx) = 2L\beta_n$$

The claim follows using the definition of the universal family of policy spaces. ■

6.2 Algorithmic Approaches

From an algorithmic point of view, the most critical part of the DPI algorithm (Figure 1) is minimizing the empirical error, which in the case of $|\mathcal{A}| > 2$ is in the following form:

$$\begin{aligned}\min_{\pi \in \Pi} \widehat{\mathcal{L}}_{\pi_k}(\widehat{\rho}; \pi) &= \min_{\pi \in \Pi} \frac{1}{N} \sum_{i=1}^{N} \Big[\max_{a \in \mathcal{A}} \widehat{Q}^{\pi_k}(x_i, a) - \widehat{Q}^{\pi_k}\big(x_i, \pi(x_i)\big) \Big] \\ &= \min_{\pi \in \Pi} \sum_{i=1}^{N} \mathbb{I}\left\{ \arg\max_{a \in \mathcal{A}} \widehat{Q}^{\pi_k}(x_i, a) \neq \pi(x_i) \right\} \Big[\max_{a \in \mathcal{A}} \widehat{Q}^{\pi_k}(x_i, a) - \widehat{Q}^{\pi_k}\big(x_i, \pi(x_i)\big) \Big].\end{aligned}$$

Unlike the two-action case, this is a multi-class cost-sensitive (MCCS) classification problem in which any classification mistake is weighted by a cost function which depends on the action taken by policy π. It is important to note that here the main difference with regression is that the goal is not to have a good approximation of the action-value function over the entire state and action space. The main objective is to have a good enough estimate of the action-value function to find the greedy action in each state. A thorough discussion on the possible approaches to MCCS classification is out of the scope of this paper, thus, we mention only a few recent methods that could be suitable for our problem. The reduction methods proposed by Beygelzimer et al. (2005, 2009) reduce the MCCS classification problem to a series of weighted binary classification problems (which can be in turn reduced to binary classification as in Zadrozny et al. 2003), whose solutions can be combined to obtain a multi-class classifier. The resulting multi-class classifier is guaranteed to have a performance which is upper-bounded by the performance of each binary classifier used in solving the weighted binary problems. Another common approach to MCCS classification is to use boosting-based methods (e.g., Lozano and Abe 2008; Busa-Fekete and Kégl 2010). Finally, a recent regression-based approach has been proposed by Tu and Lin (2010), which reduces the MCCS classification to a one-sided regression problem that can be effectively solved by a variant of SVM.

7. Conclusions

In this paper, we presented a variant of the classification-based approach to approximate policy iteration (API) called direct policy iteration (DPI) and provided its finite-sample performance bounds. To the best of our knowledge, this is the first complete finite-sample analysis for this class of API algorithms. The main difference of DPI with the existing

classification-based API algorithms (Lagoudakis and Parr, 2003b; Fern et al., 2004) is in weighting each classification error by its actual regret, i.e., the difference between the action-values of the greedy action and the action selected by DPI. Our results extend the only theoretical analysis of a classification-based API algorithm (Fern et al., 2006) by **1)** having a performance bound for the full API algorithm instead of being limited to one step policy update, **2)** considering any policy space instead of finite class of policies, and **3)** deriving a bound which does not depend on the Q-advantage, i.e., the minimum Q-value gap between a greedy and a sub-greedy action over the state space, which can be arbitrarily small in a large class of MDPs. Note that the final bound in Fern et al. (2006) depends inversely on the Q-advantage. We also analyzed the consistency of DPI and showed that although it is not consistent in general, it is consistent for the class of Lipschitz MDPs. This is similar to the consistency results for fitted value iteration in Munos and Szepesvári (2008).

One of the main motivations of this work is to have a better understanding of how the classification-based API methods can be compared with their widely-used regression-based counterparts. It is interesting to note that the bound of Equation 7 shares the same structure as the error bounds for the API algorithm in Antos et al. (2008) and the fitted value iteration in Munos and Szepesvári (2008). The error at each iteration can be decomposed into an approximation error, which depends on the MDP and the richness of the hypothesis space – the inherent greedy error in Equation 7 and the inherent Bellman error in Antos et al. (2008) and Munos and Szepesvári (2008), and an estimation error which mainly depends on the number of samples and rollouts. The difference between the approximation error of the two approaches depends on how well the hypothesis space fits the MDP at hand. This confirms the intuition that whenever the policies generated by policy iteration are easier to represent and learn than their value functions, a classification-based approach can be preferable to regression-based methods.

Possible directions for future work are:

- *The classification problem:* As discussed in Section 6.2 the main issue in the implementation of DPI is the solution of the multi-class cost-sensitive classification problem at each iteration. Although some existing algorithms might be applied to this problem, further investigation is needed to identify which one is better suited for DPI. In particular, the main challenge is to solve the classification problem without first solving a regression problem on the cost function which would eliminated the main advantage of classification-based approaches (i.e., no approximation of the action-value function over the whole state-action space).

- *Rollout allocation:* In DPI, the rollout set is build with states drawn i.i.d. from an arbitrary distribution and the rollouts are performed the same number of times for each action in $\mathcal{A}$. A significant advantage could be obtained by allocating resources (i.e., the rollouts) to regions of the state space and to actions whose action-values are more difficult to estimate. This would result in a more accurate training set for the classification problem and a better approximation of the greedy policy at each iteration. Although some preliminary results in Dimitrakakis and Lagoudakis (2008b) and Gabillon et al. (2010) show encouraging results, a full analysis of what is the best allocation strategy of rollouts over the state-action space is still missing.

Acknowledgments

This work was supported by French National Research Agency (ANR) through the projects EXPLO-RA n° ANR-08-COSI-004 and LAMPADA n° ANR-09-EMER-007, by Ministry of Higher Education and Research, Nord-Pas de Calais Regional Council and FEDER through the "contrat de projets état region (CPER) 2007–2013", and by PASCAL2 European Network of Excellence.

References

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

J. Bagnell, S. Kakade, A. Ng, and J. Schneider. Policy search by dynamic programming. In *Proceedings of Advances in Neural Information Processing Systems 16*. MIT Press, 2003.

S. Ben-David, N. Cesa-Bianchi, D. Haussler, and P. M. Long. Characterizations of learnability for classes of {0...n}-valued functions. *Journal of Computer and System Sciences*, 50:74–86, 1995.

A. Beygelzimer, V. Dani, T. Hayes, J. Langford, and B. Zadrozny. Error limiting reductions between classification tasks. In *Proceedings of the Twenty-Second International Conference on Machine Learning*, pages 49–56, 2005.

A. Beygelzimer, J. Langford, and P. Ravikumar. Error-correcting tournaments. *CoRR*, abs/0902.3176, 2009.

S. Bradtke and A. Barto. Linear least-squares algorithms for temporal difference learning. *Journal of Machine Learning*, 22:33–57, 1996.

R. Busa-Fekete and B. Kégl. Fast boosting using adversarial bandits. In *Proceedings of the Twenty-Seventh International Conference on Machine Learning*, pages 49–56, 2010.

L. Devroye, L. Györfi, and G. Lugosi. *A Probabilistic Theory of Pattern Recognition*. Springer-Verlag, 1996.

C. Dimitrakakis and M. Lagoudakis. Algorithms and bounds for sampling-based approximate policy iteration. In *Recent Advances in Reinforcement Learning (EWRL-2008)*. Springer, 2008a.

C. Dimitrakakis and M. Lagoudakis. Rollout sampling approximate policy iteration. *Machine Learning Journal*, 72(3):157–171, 2008b.

A. M. Farahmand, R. Munos, and Cs. Szepesvári. Error propagation for approximate policy and value iteration. In *Advances in Neural Information Processing Systems*, 2010.

A. Fern, S. Yoon, and R. Givan. Approximate policy iteration with a policy language bias. In *Proceedings of Advances in Neural Information Processing Systems 16*, 2004.

A. Fern, S. Yoon, and R. Givan. Approximate policy iteration with a policy language bias: Solving relational Markov decision processes. *Journal of Artificial Intelligence Research*, 25:85–118, 2006.

Victor Gabillon, Alessandro Lazaric, and Mohammad Ghavamzadeh. Rollout allocation strategies for classification-based policy iteration. In *ICML 2010 Workshop on Reinforcement Learning and Search in Very Large Spaces*, 2010.

R. A. Howard. *Dynamic Programming and Markov Processes*. The MIT Press, Cambridge, MA, 1960.

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

M. Lagoudakis and R. Parr. Reinforcement learning as classification: Leveraging modern classifiers. In *Proceedings of the Twentieth International Conference on Machine Learning*, pages 424–431, 2003b.

J. Langford and B. Zadrozny. Relating reinforcement learning performance to classification performance. In *Proceedings of the Twenty-Second international conference on Machine learning*, pages 473–480, 2005.

L. Li, V. Bulitko, and R. Greiner. Focus of attention in reinforcement learning. *Journal of Universal Computer Science*, 13(9):1246–1269, 2007.

A. Lozano and N. Abe. Multi-class cost-sensitive boosting with p-norm loss functions. In *Proceeding of the Fourteenth ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 506–514, 2008.

R. Munos. Performance bounds in Lp norm for approximate value iteration. *SIAM Journal of Control and Optimization*, 2007.

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

D. Pollard. *Convergence of Stochastic Processes*. Springer-Verlag, 1984.

H. Tu and H. Lin. One-sided support vector regression for multiclass cost-sensitive classification. In *Proceedings of the Twenty-Seventh International Conference on Machine learning*, pages 49–56, 2010.

B. Zadrozny, J. Langford, and N. Abe. Cost-sensitive learning by cost-proportionate example weighting. In *Proceedings of the Third IEEE International Conference on Data Mining*, page 435, 2003.