Robust Exploration with Tight Bayesian Plausibility Sets

UNH

Abstract

Optimism about the poorly understood states and actions is the main driving force of exploration for many provably-efficient reinforcement learning algorithms. We propose optimism in the face of sensible value functions (OFVF)- a novel reinforcement learning algorithm designed to explore robustly minimizing the worst case exploration cost. OFVF proceeds in an episodic manner, where the duration of the episode is fixed and known. OFVF relaxes the requirement for the set of plausible MDPs to be represented by a confidence interval. It also optimizes the size and location of the plausibility set. Our algorithm is inherently Bayesian and can leverage prior information. Our theoretical analysis shows the robustness of OFVF, and the empirical results demonstrate its practical promise.

1 Introduction

Markov decision processes (MDPs) provide a versatile methodology for modeling dynamic decision problems under uncertainty [Bertsekas and Tsitsiklis, 1996; Sutton and Barto, 1998; Puterman, 2005]. A perfect MDP model for many reinforcement learning problems is not known precisely in general. Instead, a reinforcement learning agent tries to maximize its cumulative payoff by interacting in an unknown environment with an effort to learn the underlying MDP model. It is important for the agent to explore sub-optimal actions to accelerate the MDP learning task which can help to optimize long-term performance. But it is also important to pick actions with highest known rewards to maximize short-run performance. So the agent always needs to balance between them to boost the performance of a learning algorithm during learning.

Optimism in the face of uncertainty (OFU) is a common principle for most reinforcement learning algorithms encouraging exploration [Jaksch *et al.*, 2010; Brafman and Tennenholtz, 2001; Kearns and Singh, 1998]. The idea is to assign a very high exploration bonus to poorly understood states and actions. The agent chooses a policy under this very "optimistic" model of the environment. As the less understood states-actions are incentivized, they seem lucrative to the agent encouraging exploration. As

the agent visits and gathers statistically significant evidence for these states-actions, the uncertainty and optimism decreases converging to reality. Many RL algorithms including *Explicit Explore or Exploit* (E^3) [Kearns and Singh, 1998], *R-MAX* [Brafman and Tennenholtz, 2001], *UCRL2* [Auer, 2006], *MBIE* [Strehl and Littman, 2008; 2004; Wiering and Schmidhuber, 1998] build on the idea of optimism guiding the exploration. These algorithms provide strong theoretical guarantees with polynomial bound on sample complexity.

The performance of these OFU algorithms greatly depends on the methods to implement optimism (e.g. Chernoff-Hoeffding's inequality for UCRL2, Confidence Interval for MBIE), which can often be complex in nature. Dealing with a family of plausible environments can sometimes become expensive as well. With OFU exploration, it is possible for an agent to be overly optimistic about a potentially catastrophic situation and end up there paying an extremely high price (e.g. a self driving car hits a wall, a robot falls off the cliff etc.). Exploring and learning such a situation may not payoff the price. It can be wise for the agent to be robust and avoid those situations minimizing the worst-case exploration cost− which we call robust exploration. OFU algorithms are optimistic by definition and cannot guarantee robustness while exploring.

Probability matching class of algorithms like *Thompson sampling* [Thompson, 1933] performs exploration with a proportional likelihood to the underlying true parameters and has been successfully applied to multi-armed bandit problems [Agrawal and Goyal, 2012a;b]. *Posterior Sampling for reinforcement learning (PSRL)* [Osband and Van Roy, 2017; Osband *et al.*, 2013; Strens, 2000] applies the same idea in the context of reinforcement learning. PSRL algorithm samples a single instance of the environment from the posterior distribution, then solves and executes the policy optimal for that sampled environment over the episode. Selection of a policy in PSRL is proportional to the probability of that being optimal and exploration is guided by the variance of sampled policies as opposed to optimism. PSRL algorithm is simple, computationally efficient and can utilize any prior structural information to improve exploration. Strong theoretical analysis and practical applications for PSRL are also established in the literature. But similar to OFU algorithms, PSRL cannot handle worst case exploration penalty and performs poorly in such situations.

The main contribution of this paper is OFVF, a new *data-driven* Bayesian approach to constructing *Plausibility* sets for MDPs. The method computes policies with tighter robust estimates for exploration by introducing two new ideas. First, it is based on Bayesian posterior distributions rather than distribution-free bounds. Second, OFVF does not construct plausibility sets as simple confidence intervals. Confidence intervals as plausibility sets are a sufficient but not a necessary condition. OFVF uses the structure of the value function to optimize the *location* and *shape* of the plausibility set to guarantee upper bounds directly without necessarily enforcing the requirement for the set to be a confidence interval.

The paper is organized as follows ...

2 Problem Statement

We consider the problem of learning a finite horizon Markov Decision Process $\mathcal{M}$ with states $\mathscr{S} = \{1, \ldots, S\}$ and actions $\mathscr{A} = \{1, \ldots, A\}$. $p : \mathscr{S} \times \mathscr{A} \to \Delta^{\mathscr{S}}$ is a transition function, where $p_{ss'}^a$ is interpreted as the probability of ending in state $s' \in \mathscr{S}$ by taking an action $a \in \mathscr{A}$ from state $s \in \mathscr{S}$. We omit s' when the next state is not deterministic and denote the transition probability as $p_{sa} \in \mathbb{R}^S$. $R : \mathscr{S} \times \mathscr{A} \to \mathbb{R}$ is a reward function and $R_{ss'}^a$ is the reward for taking action $a \in \mathscr{A}$ from state $s \in \mathscr{S}$ and reaching state $s' \in \mathscr{S}$. Each MDP $\mathcal{M}$ is associated with a discount factor $0 \leqslant \gamma \leqslant 1$ and a distribution of initial state probabilities p_0. L is the number of episodes and H is the number of periods in each episode, we consider an episodic learning environment throughout the paper. In each episode $l \in L$, an initial state $s_0 \in \mathscr{S}$ is sampled from p_0. In period $h = 0 \ldots H$ of each episode, for a state s_h and action a_h, a next state s_{h+1} is sampled following $p_{s_h s_{h+1}}^{a_h}$ and a reward is obtained from $R_{s_h s_{h+1}}^{a_h}$.

A policy $\pi = (\pi_0, \ldots, \pi_{H-1})$ is a set of functions mapping a state $s \in \mathscr{S}$ to an action $a \in \mathscr{A}$. We define a value function for each policy π as:

$$V_h^\pi(s) := \sum_{s'} P_{ss'}^{\pi(s)}[r_h + V(s')] \tag{1}$$

The optimal value function is defined by $V_h^\star(s) = \max_\pi V_h^\pi(s)$ and the optimal policy is defined by $\pi^\star(s) = \arg\max_{a \in \mathscr{A}} p_{ss'}^a V(s'), \forall s' \in \mathscr{S} : p_{ss'}^a > 0$. We also define the state-action optimal value function for $h = 0, \ldots, H-1$ as:

$$Q_h^\star(s, a) := \sum_{s'} P_{ss'}^a[r_h + V^\star(s')] \tag{2}$$

The optimal policy $\pi^\star$ is defined as $\pi^\star(s) = \arg\max_{a \in \mathscr{A}} Q_h^\star(s, a), \forall s, h$. Optimistic algorithms encouraging exploration find the probability distribution $\tilde{P}_{sa}$ for each state and action within an interval of the

empirically derived distribution $\bar{p}_{sa} = \mathbb{E}[\cdot|s, a]$, which defines the plausible set $\mathscr{P}_{sa}$ of MDPs. They then solve an optimistic version of Eq. (2) within $\mathscr{P}_{sa}$ that leads to the policy with highest reward.

$$Q_h^\star(s, a) := \max_{p^{sa} \in \mathscr{P}^{sa}} \sum_{s'} p_{ss'}^a[r_h + V^\star(s')] \tag{3}$$

An RL agent interacts with the environment in an episodic setting. At each episode $l \in 0, \ldots, L-1$, the agent takes actions based on the policy $\pi_l^\star$ optimal in episode l and realizes reward $\sum_{h=0}^H r_{lh}$. $\pi_l^\star$ is computed from the experiences gathered by the agent in previous episodes $0, \ldots, l-1$. We evaluate the performance of the agent in terms of *expected cumulative regret*:

$$Regret(T) = \sum_{l=0}^{T/H-1} \mathbb{E}\left[V^\star(s_0) - V^{\pi_l^\star}(s_0)\right]$$

Where $s_0 \sim p_0$, and $V^\star(s_0)$ are the true values of initial states..

3 Interval Estimation for Plausibility Sets

When solving an MDP in reinforcement learning, the main type of uncertainties are encountered on the model parameters, namely in transition probabilities and rewards. Given data set $\mathscr{D}$, an interval estimation of the model parameters is able to acknowledge and quantify this uncertainty, which a maximum likelihood based point estimate cannot do. In this section, we first describe the standard approach to constructing plausibility sets as distribution free confidence intervals. We then propose its extension to Bayesian setting and present a simple algorithm to serve that purpose. It is important to note that distribution-free bounds are subtly different from the Bayesian bounds, the Bayesian safety guarantee holds conditional on a given dataset $\mathscr{D}$ while the distribution-free hold across the sets $\mathscr{D}$. This makes the guarantees qualitatively different and difficult to compare.

3.1 Plausibility Sets as Confidence Intervals

It is common in the literature to use L_1 norm as the distribution-free bound. This bound is constructed around the empirical mean of the transition probability $\bar{P}_{s,a}$ by applying the Hoeffding inequality [Auer *et al.*, 2009; 2010; Petrik and Luss, 2016; Wiesemann, 2013; Strehl and Littman, 2004].

$$\mathscr{P}^{sa} = \left\{\|\tilde{p}_{sa} - \bar{p}_{sa}\|_1 \leqslant \sqrt{\frac{2}{n_{s,a}} \log \frac{SA2^S}{\delta}}\right\}$$

where $\bar{P}_{sa}$ is the mean transition computed from D, $n_{s,a}$ is the number of times the agent arrived state s' after taking action a in state s, δ is the required probability of the interval and $\|\cdot\|_1$ is the L_1 norm. An important limitation of

this approach is that, the size of $\mathscr{P}_{sa}$ grows linearly with the number of states, which makes it practically useless in general.

3.2 Bayesian Plausibility Sets

The Bayesian plausibility sets take the same interval estimation idea and extend it into Bayesian setting, which is analogous to *credible intervals* in Bayesian statistics. Credible intervals are constructed with the posterior probability distributions and they are fixed − not a random variable, given the data $\mathscr{D}$. Instead the estimated transition probabilities maximizing the rewards are random variables. They can take advantage of situation-specific prior knowledge unlike general confidence intervals as discussed above. However, a concise comparison between Bayesian and frequentist bounds is beyond the scope of this paper.

The credible region for the plausibility set can be constructed around the mean, median or mode of the sample posterior distribution. Among which, bound around the empirical mean of the transition probabilities is simple and more common. To construct such a region, we optimize for the smallest plausibility set around the mean transition probability with the assumption that a smaller plausibility set will lead to a tighter upper bound estimate. Formally, the optimization problem to compute $\psi_{s,a}$ for each state s and action a is:

$$\min_{\psi \in \mathbb{R}_+} \{\psi \ : \ \mathbb{P}\left[\|\tilde{p}_{s,a} - \bar{p}_{s,a}\|_1 > \psi \mid \mathscr{D}\right] < \delta\} \ , \quad (4)$$

where nominal point is $\bar{p}_{s,a} = \mathbb{E}_{\tilde{P}}[\tilde{p}_{s,a} \mid \mathscr{D}]$. The nominal point is not included in the optimization to reduce computational complexity.

The optimization problem in Eq. (4) can be solved by the Sample Average Approximation (SAA) algorithm [Shapiro *et al.*, 2014]. The main idea is to sample from the posterior distribution and then choose the minimal size $\psi_{s,a}$ that satisfies the constrain. Algorithm 3, in the appendix, summarizes the simple sort-based method.

3.3 Bayesian Upper Confidence Reinforcement Learning

We are now ready to describe a simple Bayesian optimistic algorithm, named as Bayes-UCRL. This builds directly on top of UCRL [Auer *et al.*, 2009; 2010], which constructs the plausibility set as distribution free bounds, similar to what is described in Section 3.1. Unlike UCRL, Bayes-UCRL constructs the plausibility set in a Bayesian setting, as described in Section 3.2. Algorithm 1 describes the idea in more detail. The algorithm uses a hierarchical Bayesian model that can be used to infer the posterior transition probability over $p^\star$. The details of the Bayesian model are largely irrelevant for our purpose. The model may have a

---

Algorithm 1: Bayes UCRL

Input: Prior distribution θ^0 over $p^\star_{s,a}$, desired confidence level δ, number of episodes K

Output: Policy with an optimistic return estimate

1   $k = 1, X = \phi$;

2 **repeat**

3   Initialize MDP: $\mathscr{M}^k$;

4   Compute posterior from θ^{k-1} and samples X: θ^k;

5   **for** *All states and actions:* $s \in \mathscr{S}, a \in \mathscr{A}$ **do**

6   Compute mean transition $\bar{p}_{s,a}$, and interval $\psi_{s,a}$ (Invoke Algorithm 3);

7   Add transition for s and a in $\mathscr{M}^k$;

8   Compute value function and policy: $v_k, \hat{\pi}_k \leftarrow$ solve $\mathscr{M}^k$;

9    $X \leftarrow$ execute $\hat{\pi}_k$ and collect samples;

10    $k \leftarrow k + 1$

11 **until** $k \leqslant K$;

12 **return** $(\hat{\pi}_k, p_0^\mathsf{T} v_k)$;

---

simple analytical posterior, or may need to be sampled using MCMC methods like Stan [Gelman *et al.*, 2014]. The algorithm assumes that it is possible to draw enough samples from the posterior that the sampling error becomes negligible.

Algorithm 1 proceeds in an episodic manner. At the beginning of each episode, it computes the posterior transition probabilities for each state and action. It then invokes Algorithm 3 to compute the nominal transition points and the interval estimations for each state and action. It solves the MDP to compute the optimal policy for current episode. The execution of this policy for a fixed horizon generates more samples for posterior estimation in the next episode. This algorithm requires some extra steps compared to UCRL. It is also computationally expensive. But it performs better than UCRL in terms of computed returns over episodes which we will see in Section 5.

4 OFVF: Optimism in the Face of sensible Value Functions

OFVF uses samples from a posterior distribution, similar to a Bayesian confidence interval, but it relaxes the safety requirement as it is sufficient to guarantee for each state s and action a that:

$$\max_{v \in \mathscr{V}} \mathbb{P}_{P^\star}\left[\min_{p \in \mathscr{P}_{s,a}} (p - p^\star_{s,a})^\mathsf{T} v \leqslant 0 \,\middle|\, \mathscr{D}\right] \geqslant 1 - \delta \ , \quad (5)$$

with $\mathscr{V} = \{\hat{v}^\star_{\mathscr{D}}\}$. To construct the set $\mathscr{P}$ here, the set $\mathscr{V}$ is not fixed but depends on the robust solution, which in turn depends on $\mathscr{P}$. RSVF starts with a guess of a small set for $\mathscr{V}$ and then grows it, each time with the current value

function, until it contains $\hat{v}^{\star}_{\mathscr{P}}$ which is always recomputed after constructing the ambiguity set $\mathscr{P}$.

Algorithm 2: OFVF: Optimism in the Face of sensible Value Functions

Input: Desired confidence level δ and posterior distribution $\mathbb{P}_{P^{\star}}[\cdot \mid \mathscr{D}]$

Output: Policy with a maximized safe return estimate

1 Initialize current policy
$\quad \pi_0 \leftarrow \arg\max_{\pi} \rho(\pi, \mathbb{E}_{P^{\star}}[P^{\star} \mid \mathscr{D}])$;
2 Initialize current value $v_0 \leftarrow v^{\pi_0}_{\mathbb{E}_{P^{\star}}[P^{\star} \mid \mathscr{D}]}$;
3 Initialize value robustness set $\mathscr{V}_0 \leftarrow \{v_0\}$;
4 Construct $\mathscr{P}_0$ optimal for $\mathscr{V}_0$;
5 Initialize counter $k \leftarrow 0$;
6 **while** *Eq. (5) is violated with $\mathscr{V} = \{v_k\}$* **do**
7 $\quad$ Include v_k that violates Eq. (5):
$\quad\quad \mathscr{V}_{k+1} \leftarrow \mathscr{V}_k \cup \{v_k\}$;
8 $\quad$ Construct $\mathscr{P}_{k+1}$ optimized for $\mathscr{V}_{k+1}$;
9 $\quad$ Compute robust value function v_{k+1} and policy π_{k+1} for $\mathscr{P}_{k+1}$;
10 $\quad$ $k \leftarrow k+1$;
11 **return** $(\pi_k, p_0^{\mathsf{T}} v_k)$;

In lines 4 and 8 of Algorithm 2, $\mathscr{P}_i$ is computed for each state-action $s, a \in \mathscr{S} \times \mathscr{A}$. Center $\bar{p}$ and set size $\psi_{s,a}$ are computed from Eq. (7) using set $\mathscr{V}$ & optimal g_v computed by solving Eq. (6). When the set $\mathscr{V}$ is a singleton, it is easy to compute a form of an optimal ambiguity set.

$$g = \max\left\{k \ : \ \mathbb{P}_{P^{\star}}[k \leqslant v^{\mathsf{T}} p^{\star}_{s,a}] \geqslant 1 - \delta/(SA)\right\} \quad (6)$$

When $\mathscr{V}$ is a singleton, it is sufficient for the ambiguity set to be a subset of the hyperplane $\{p \in \Delta^S \ : \ v^{\mathsf{T}} p = g^{\star}\}$ for the estimate to be safe. When $\mathscr{V}$ is not a singleton, we only consider the setting when it is discrete, finite, and relatively small. We propose to construct a set defined in terms of an L_1 ball with the minimum radius such that it is safe for every $v \in \mathscr{V}$. Assuming that $\mathscr{V} = \{v_1, v_2, \ldots, v_k\}$, we solve the following linear program:

$$\psi_{s,a} = \min_{p \in \Delta^S}\left\{\max_{i=1,\ldots,k} \|q_i - p\|_1 \ : \ v_i^{\mathsf{T}} q_i = g_i^{\star}, q_i \in \Delta^S, i \in 1, \ldots, k\right\} \quad (7)$$

In other words, we construct the set to minimize its radius while still intersecting the hyperplane for each v in $\mathscr{V}$. Algorithm 2, as described, is not guaranteed to converge in finite time as written. It can be readily shown the value functions in the individual iterations are non-increasing. It is easy to just stop once the value function becomes smaller (and that is more conservative) than BCI.
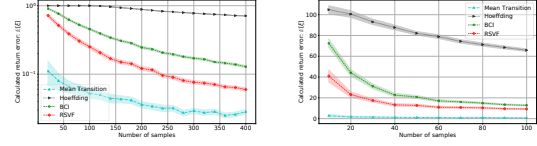

Figure 1: Return error with a Gaussian prior with 95% confidence, Left: Single state, Right: Full MDP, X-axis is the number of samples per state-action.

5 Empirical Evaluation

In this section, we evaluate the worst-case estimates computed by Bayes UCRL and OFVF empirically. We assume a true model of each problem and generate a number of simulated data sets for the known distribution. We compute the tightest optimistic estimate for the optimal return and compare it with the optimal return for the true model. We compare our results with "UCRL2" and "PSRL" algorithms. The value ξ represents the predicted regret, which is the absolute difference between the *true* optimal value and the robust estimate: $\xi = |\rho(\pi^{\star}_{P^{\star}}, P^{\star}) - \hat{\rho}(\hat{\pi}^{\star})|$, a smaller regret is better. All of our experiments use a 95% confidence for safety unless otherwise specified.

5.1 Single-state Bellman Update

We initially consider simple problems where transition from a single non-terminal state following a single action leads to multiple terminal states. The value function for the terminal states are fixed and assumed to be provided. We evaluate different priors over the transition probabilities: i) uninformative Dirichlet prior and ii) informative Gaussian prior. Note that RSVF is optimal in this simplistic setting, as **??** (left) and Fig. 1 (left) shows. As expected, the mean estimate provides the tightest bound, but **??** (right) illustrates that it does not provide any meaningful safety guarantees.

5.2 River Swim Problem

5.3 Mountain Car Problem

References

S. Agrawal and N. Goyal. Thompson Sampling for contextual bandits with linear payoffs. *Proceedings of the 30th International Conference on Machine Learning (ICML)*, 28:127–135, 2012.

Shipra Agrawal and N Goyal. Analysis of Thompson sampling for the multi-armed bandit problem. In *Annual Conference on Learning Theory (COLT)*, pages 39.1–39.26, 2012.

Peter Auer, Thomas Jaksch, and Ronald Ortner. Near-

optimal Regret Bounds for Reinforcement Learning. In D Koller, D Schuurmans, Y Bengio, and L Bottou, editors, *Advances in Neural Information Processing Systems 21*. 2009.

P Auer, Thomas Jaksch, and R Ortner. Near-optimal regret bounds for reinforcement learning. *Journal of Machine Learning Research*, 11(1):1563–1600, 2010.

Peter Auer. Logarithmic Online Regret Bounds for Undiscounted Reinforcement Learning. *Advances in Neural Information Processing Systems (NIPS)*, 2006.

Dimitri P Bertsekas and John N Tsitsiklis. *Neuro-dynamic programming*. 1996.

Ronen I. Brafman and Moshe Tennenholtz. R-MAX - A general polynomial time algorithm for near-optimal reinforcement learning. *International Joint Conference on Artificial Intelligence (IJCAI)*, 2001.

Andrew Gelman, John B Carlin, Hal S Stern, and Donald B Rubin. *Bayesian Data Analysis*. 3rd edition, 2014.

Thomas Jaksch, Ronald Ortner, and Peter Auer. Near-optimal Regret Bounds for Reinforcement Learning. *Journal of Machine Learning Research*, 11(1):1563–1600, 2010.

Michael Kearns and Satinder Singh. Near-optimal reinforcement learning in polynomial time. In *International Conference on Machine Learning*, volume 49, pages 260–268. Morgan Kaufmann, 1998.

Ian Osband and Benjamin Van Roy. Why is Posterior Sampling Better than Optimism for Reinforcement Learning? *International Conference on Machine Learning (ICML)*, 2017.

Ian Osband, Daniel Russo, and Benjamin Van Roy. (More) Efficient Reinforcement Learning via Posterior Sampling? *Advances in Neural Information Processing Systems (NIPS)*, 2013.

Marek Petrik and Ronny Luss. Interpretable Policies for Dynamic Product Recommendations. In *Uncertainty in Artificial Intelligence (UAI)*, 2016.

Martin L Puterman. *Markov decision processes: Discrete stochastic dynamic programming*. John Wiley & Sons, Inc., 2005.

A. Shapiro, D. Dentcheva, and A. Ruszczynski. *Lectures on stochastic programming: Modeling and theory*. 2014.

a. L. Strehl and M. L. Littman. An empirical evaluation of interval estimation for markov decision processes. (April 2007):128–135, 2004.

Alexander L Strehl and Michael L Littman. *Journal of Computer and System Sciences*, 74:1309–1331, 2008.

Malcolm Strens. A Bayesian Framework for Reinforcement Learning. *International Conference on Machine Learning (ICML)*, 2000.

Richard S Sutton and Andrew Barto. *Reinforcement learning*. 1998.

W.R. Thompson. On the Likelihood that One Unknown Probability Exceeds Another in View of the Evidence of Two Samples. *Oxford University Press*, 25(3):285–294, 1933.

Marco Wiering and Jurgen Schmidhuber. *International Conference on Simulation of Adaptive Behavior (SAB)*, pages 223–228, 1998.

Wolfram Wiesemann. Robust Markov decision processes. *. . . of Operations Research*, pages 1–52, 2013.

A Technical Results

A.1 Computing Bayesian Confidence Interval

Algorithm 3: Bayesian Confidence Interval (BCI)

Input: Distribution θ over $p_{s,a}^{\star}$, confidence level δ, sample count m

Output: Nominal point $\bar{p}_{s,a}$ and L_1 norm size $\psi_{s,a}$

1 Sample $X_1, \ldots, X_m \in \Delta^S$ from θ: $X_i \sim \theta$;

2 Nominal point: $\bar{p}_{s,a} \leftarrow (1/m) \sum_{i=1}^{m} X_i$;

3 Compute distances $d_i \leftarrow \|\bar{p}_{s,a} - X_i\|_1$ and sort *increasingly*;

4 Norm size: $\psi_{s,a} \leftarrow d_{(1-\delta)\,m}$;

5 **return** $\bar{p}_{s,a}$ *and* $\psi_{s,a}$;