Learning Adversarial Low-rank MDPs with Unknown Transition and Full-information Feedback

Canzhe Zhao, Ruofeng Yang, Baoxiang Wang, Xuezhou Zhang, Shuai Li

上海交通大学
约翰·霍普克罗夫特
计算机科学中心
John Hopcroft Center for Computer Science

香港中文大學（深圳）
The Chinese University of Hong Kong, Shenzhen

BOSTON UNIVERSITY

Contributions

- The first algorithm for learning adversarial low-rank MDPs, called **P**olicy **O**ptimization for **LO**w-rank MDPs (**POLO**), that simultaneously tackles the representation learning and adversarially changed loss functions in RL.
- Attains the $\widetilde{O}(K^{5/6}A^{1/2}d\ln(1+M)/(1-\gamma)^2)$ regret upper bound.
- An $\Omega(\frac{\gamma^2}{1-\gamma}\sqrt{dAK})$ regret lower bound is also provided, serving as the first regret lower bound for learning low-rank MDPs in the regret minimization setting.

Setting

Episodic Infinite-horizon Adversarial MDPs $(\mathcal{S}, \mathcal{A}, P^\star, \{\ell_k\}_{k=1}^K, \gamma, d_0)$

- $\mathcal{S}$ and $\mathcal{A}$: state and action spaces
- $P^\star : \mathcal{S} \times \mathcal{A} \times \mathcal{S} \to [0,1]$: transition probability kernel
- $\gamma \in [0,1)$: discount factor
- $d_0 \in \Delta(\mathcal{S})$: initial distribution over state space
- $\ell_k : \mathcal{S} \times \mathcal{A} \to [0,1]$: loss function in episode k

Low-rank MDPs An MDP is a low-rank MDP if there exist two feature embedding functions $\phi^\star : \mathcal{S} \times \mathcal{A} \to \mathbb{R}^d$, $\mu^\star : \mathcal{S} \to \mathbb{R}^d$ such that for any $(s,a,s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}$, $P^\star(s' \mid s,a) = \mu^\star(s')^\top \phi^\star(s,a)$, where $\|\phi^\star(s,a)\|_2 \le 1$ and for any function $g : \mathcal{S} \to [0,1]$, $\left\|\int \mu^\star(s)g(s)\mathrm{d}(s)\right\|_2 \le \sqrt{d}$.

Learning Objective Minimize the *pseudo regret* with respect to $\pi^\star$, defined as $\mathcal{R}_K = \mathbb{E}\left[\sum_{k=1}^K \left(V_k^{\pi_k} - V_k^{\pi^\star}\right)\right]$, where $\pi^\star \in \mathrm{argmin}_{\pi\in\Pi}\mathbb{E}\left[\sum_{k=1}^K V_k^\pi\right]$ is the fixed optimal policy in hindsight and Π is the set of all stochastic policies.

Algorithm

Doubled Exploration and Exploitation

- *One-step trick* to guarantee the (near) optimism of the estimated value functions at d_0 in [3]:

$$\mathbb{E}_{(s,a)\sim d_{P^\star}^{\tilde{\pi}}}[g(s,a)] \le (1-\gamma)^{-1}\mathbb{E}_{(s,a)\sim d_{P^\star}^{\tilde{\pi}}}\left[\|\phi^\star(s,a)\|_{\Sigma^{-1}_{\rho_k,\phi^\star}}\right]\sqrt{k\gamma A\mathbb{E}_{\rho_k'}\left[g^2(s,a)\right] + \gamma\lambda_k dB^2}\,,$$

where $\rho_k(s,a) = 1/k\sum_{i=1}^k d_{P^\star}^{\tilde{\pi}_i}(s,a)$ and $\rho_k'(s,a) = 1/k\sum_{i=1}^k d_{P^\star}^{\tilde{\pi}_i}(s)U(a)$.
- Two-step exploration by sampling actions from $U(\cdot)$ after collecting $s_k \sim d_{P^\star}^{\tilde{\pi}_k}$.
- Previous algorithms (*e.g.*, algorithm in [3]) have no regret guarantees due to the uniform exploration, even in the stochastic setting.
- Instead, our POLO uses a mixed roll-out policy to interleave (a) the exploration over transitions required by representation learning; and (b) the exploration and exploitation over the adversarial loss functions by policy optimization.
- Formally, conducts the exploration over the transitions with probability ξ and execute policy $\tilde{\pi}_k$ optimized by online mirror descent (OMD) with probability $1-\xi$, respectively.

Empirical Model Update

- Performing maximum likelihood estimation (MLE) over the updated datasets to obtain the empirical transition $\widehat{P}_k$ by solving $\left(\widehat{\mu}_k, \widehat{\phi}_k\right) = \underset{(\mu,\phi)\in\mathcal{M}}{\mathrm{argmax}}\,\mathbb{E}_{\mathcal{D}_k\cup\mathcal{D}_k'}\left[\ln \mu^\top(s')\phi(s,a)\right]$, where $\mathbb{E}_{\mathcal{D}}\left[f(s,a,s')\right] = 1/|\mathcal{D}|\sum_{(s,a,s')\in\mathcal{D}} f(s,a,s')$.

Algorithm 1 Policy Optimization for Low-rank MDPs (POLO)

Input: Mixing coefficient ξ, epoch length L, regularization coefficients $\{\lambda_k\}_{k=1}^K$, bonus coefficients $\{\alpha_k\}_{k=1}^K$, model class $\mathcal{M}$, number of episodes K, learning rate η.
Initialization: Set $\mathcal{D}_0 = \emptyset, \mathcal{D}_0' = \emptyset$.
for $i = 1, 2, \dots, \lceil K/L \rceil$ **do**
  Set $k_i = (i-1)L + 1$ and $\tilde{\pi}_{k_i}(\cdot \mid s)$ to be uniform for any $s \in \mathcal{S}$.
  for $k = k_i, k_i+1, \dots, k_i+L-1$ **do**
    Sample s_k from $d_{P^\star}^{\tilde{\pi}_k}$.
    Sample $c_k \sim \mathrm{Ber}(1-\xi)$.
    if $c_k = 1$ **then**
      Sample $a_k \sim \tilde{\pi}_k(\cdot \mid s_k), s_k' \sim P^\star(\cdot \mid s_k, a_k), a_k' \sim \tilde{\pi}_k(\cdot \mid s_k'), s_k'' \sim P^\star(\cdot \mid s_k', a_k')$.
    else
      Sample $a_k \sim U(\mathcal{A}), s_k' \sim P^\star(\cdot \mid s_k, a_k), a_k' \sim U(\mathcal{A}), s_k'' \sim P^\star(\cdot \mid s_k', a_k')$.
    end if
    Observe the loss function ℓ_k.
    Update datasets $\mathcal{D}_k = \mathcal{D}_{k-1} \cup \{(s_k, a_k, s_k')\}, \mathcal{D}_k' = \mathcal{D}_{k-1}' \cup \{(s_k', a_k', s_k'')\}$.
    if $k = k_i$ **then**
      Set the empirical transition $\widehat{P}_k(s' \mid s,a) = \widehat{\mu}_k(s')^\top\widehat{\phi}_k(s,a), \forall (s,a,s') \in \mathcal{S} \times \mathcal{A} \times \mathcal{S}$ via solving Eq. (1).
      Update the empirical covariance matrix $\widehat{\Sigma}_k = \sum_{(s,a)\in\mathcal{D}_k} \widehat{\phi}_k(s,a)\widehat{\phi}_k(s,a)^\top + \lambda_k I$.
      Set the bonus function $\widehat{b}_k(s,a) := \min(\alpha_k\|\widehat{\phi}_k(s,a)\|_{\widehat{\Sigma}_k^{-1}}, 2)/(1-\gamma), \forall (s,a) \in \mathcal{S} \times \mathcal{A}$
    else
      Set the empirical transition $\widehat{P}_k = \widehat{P}_{k_i}$ and bonus function $\widehat{b}_k = \widehat{b}_{k_i}$.
    end if
    Compute $\widehat{Q}_k^{\tilde{\pi}_k}(\cdot,\cdot) = \text{Policy-Evaluation}(\widehat{P}_k, \ell_k - \widehat{b}_k, \tilde{\pi}_k)$.
    Update policy $\tilde{\pi}_{k+1}(\cdot \mid \cdot) \propto \tilde{\pi}_k(\cdot \mid \cdot)\exp(-\eta\widehat{Q}_k^{\tilde{\pi}_k}(\cdot,\cdot))$.
  end for
end for

Policy Optimization in Fixed Learned Models

- Previous OMD-based PO methods for tabular and linear (mixture) MDPs [2, 1] critically depend on the *point-wise optimism* for each state-action pair, *i.e.*, $\widehat{Q}_k^{\tilde{\pi}_k}(s,a) \le \ell_k(s,a) + \gamma[P^\star\widehat{V}_k^{\tilde{\pi}_k}](s,a)$, to enable the decomposition (*cf.*, Lemma 1 by [2])

$$\widehat{V}_k^{\tilde{\pi}_k}(s_0) - V_k^{\pi^\star}(s_0) = \mathbb{E}\left[\sum_{\tau=0}^{\infty}\gamma^\tau\left\langle\tilde{\pi}_k(\cdot \mid s_\tau) - \pi^\star(\cdot \mid s_\tau), \widehat{Q}_k^{\tilde{\pi}_k}(s_\tau,\cdot)\right\rangle\middle|\,\pi^\star, P^\star, s_0\right]$$
$$+\,\mathbb{E}\left[\sum_{\tau=0}^{\infty}\gamma^\tau\left(\widehat{Q}_k^{\tilde{\pi}_k}(s_\tau,a_\tau) - \ell_k(s_\tau,a_\tau) - \gamma\left[P^\star\widehat{V}_k^{\tilde{\pi}_k}\right](s_\tau,a_\tau)\right)\middle|\,\pi^\star, P^\star, s_0\right],$$

where $\widehat{Q}_k^{\tilde{\pi}_k}$ is the Q value function of $\tilde{\pi}_k$ on $(\widehat{P}_k, \ell_k - \widehat{b}_k)$ with $\widehat{b}_k$ as some bonus function.
- The first term is contributed by competing with $\pi^\star$ in the *true* model $P^\star \Longrightarrow$ conducting policy optimization in the *true* model.
- Unfortunately, not applicable in low-rank MDPs, due to the unknown representations.
- Instead, we consider the following decomposition:

$$\begin{aligned}&\widehat{V}_k^{\tilde{\pi}_k}(s_0) - V_k^{\pi^\star}(s_0)\\ =&\widehat{V}_k^{\tilde{\pi}_k}(s_0) - \widehat{V}_k^{\pi^\star}(s_0) + \widehat{V}_k^{\pi^\star}(s_0) - V_k^{\pi^\star}(s_0)\\ =&\mathbb{E}\left[\sum_{\tau=0}^{\infty}\gamma^\tau\left\langle\tilde{\pi}_k(\cdot \mid s_{k,\tau}) - \pi^\star(\cdot \mid s_{k,\tau}), \widehat{Q}_k^{\tilde{\pi}_k}(s_{k,\tau},\cdot)\right\rangle\middle|\,\pi^\star, \widehat{P}_k, s_0\right] + \widehat{V}_k^{\pi^\star}(s_0) - V_k^{\pi^\star}(s_0)\,,\end{aligned}$$

- The first term is contributed by competing against $\pi^\star$ in the *learned* model $\widehat{P}_k \Longrightarrow$ conducting policy optimization in *learned* models.
- This decomposition will be amenable as long as we can achieve a *near optimism* at the initial state s_0, *i.e.*, $\widehat{V}_k^{\pi^\star}(s_0) - V_k^{\pi^\star}(s_0) \lesssim 0$
- **Caveat**: the local update nature of PO at each state + state occupancy distribution $d_{\widehat{P}_k}^{\pi^\star}$ varies across different episodes $\Longrightarrow$ the first term above is no longer bounded by OMD analysis!
- To address this issue, POLO adopts an epoch-based transition update, in which one epoch has L episodes and the model is only updated at the first episode in one epoch.
- With $D_F(x,y)$ as the KL divergence, at the end of episode k, the policy is updated by solving

$$\tilde{\pi}_{k+1}(\cdot \mid s) \in \mathrm{argmin}_{\pi(\cdot|s)\in\Delta(\mathcal{A})}\eta\left\langle\pi(\cdot \mid s), \widehat{Q}_k^{\tilde{\pi}_k}(s,\cdot)\right\rangle + D_F(\pi(\cdot \mid s), \tilde{\pi}_k(\cdot \mid s))\,.$$

Analysis

Regret Upper Bound

Theorem 1. *For any adversarial low-rank MDP, with appropriate setting of parameters, the regret of POLO is upper bounded by* $\mathcal{R}_K = O(K^{5/6}A^{1/2}d\ln\left(1 + AMK^2\right)/(1-\gamma)^2)$.

Remark. *Ignoring the dependence on all logarithmic factors but* M*, the regret upper bound can be simplified as* $\widetilde{O}(K^{5/6}A^{1/2}d\ln(1+M)/(1-\gamma)^2)$. *The regret upper bound matches the regret lower bound* $\Omega(\frac{\gamma^2}{1-\gamma}\sqrt{dAK})$ *in* A *up to a logarithmic factor but looses in factors of* K *and* d.

Regret Lower Bound

Theorem 2. *Suppose* $d \ge 8$, $S \ge d+1$, $A \ge d-3$, *and* $K \ge 2(d-4)A$. *Then for any algorithm* Alg, *there exists an episodic infinite-horizon low-rank MDP* $\mathcal{M}_{\mathrm{Alg}}$ *with fixed loss function such that the regret for this MDP is lower bounded by* $\Omega(\frac{\gamma^2}{1-\gamma}\sqrt{dAK})$.

Remark. • *The first regret lower bound for learning low-rank MDPs with fixed loss functions.*
- *The dependence on* A *in Theorem 2 shows a clear separation between low-rank MDPs and linear MDPs, which demonstrates that low-rank MDPs are statistically more difficult to learn than linear MDPs in the regret minimization setting.*

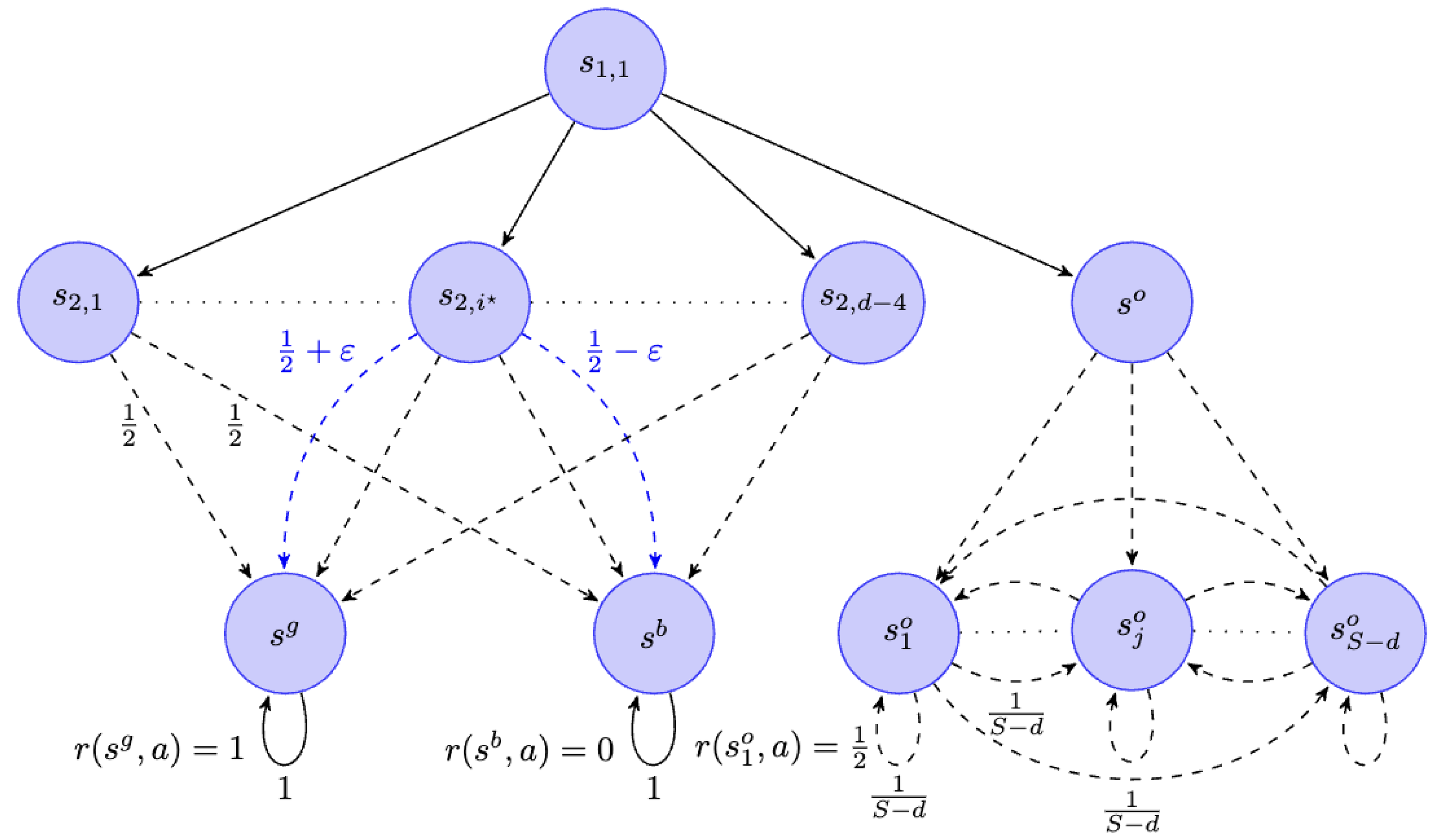

Figure 1: The class of the hard-to-learn low-rank MDP instances used in the proof of Theorem 2.

References

[1] Jiafan He, Dongruo Zhou, and Quanquan Gu. Near-optimal policy optimization algorithms for learning adversarial linear mixture mdps. In *AISTATS 2022*.

[2] Lior Shani, Yonathan Efroni, Aviv Rosenberg, and Shie Mannor. Optimistic policy optimization with bandit feedback. In *ICML 2020*.

[3] Masatoshi Uehara, Xuezhou Zhang, and Wen Sun. Representation learning for online and offline RL in low-rank mdps. In *ICLR 2022*.