# Provable benefits of low rank representation in multi-task offline learning: a strategy based on convex optimization

**Lei Shi**
Division of Biostatistics, School of Public Health
University of California, Berkeley
leishi@berkeley.edu

**Tianhao Wu**
Department of Electrical Engineering and Computer Sciences
University of California, Berkeley
thw@berkeley.edu

## Abstract

This paper provides theoretical insights into offline reinforcement learning combined with multi-task representation learning. When a collection of relevant reinforcement learning tasks are available, we propose an offline actor-critic algorithm that can take advantages of the low dimensional shared latent structure and aggregate information across different tasks to improve statistical efficiency. Under several assumptions such as linear function approximation and the closeness of Bellman evaluation operator, we derive an upper bound on the joint suboptimality gap of the policy learned by the proposed algorithm, which is characterized by both the *optimization error* and the *Uncertainty error*. This upper bound implies that incorporating latent feature representation learning can improve the regret bound significantly when the dimension $k$ of the inherent structure is much smaller than the number of tasks $M$ and the dimension of shared features $d$ given that the coverage for every tasks is uniformly good.

## 1 Introduction

Offline reinforcement learning (RL) has received wide study and significant development in recent years. It is especially beneficial in settings where online interaction is impractical, either because data collection is expensive (e.g., in robotics, educational agents, etc. Kober et al. [2013], Bassen et al. [2020]) or dangerous (e.g., in autonomous driving, or healthcare, etc. Sallab et al. [2017], Gottesman et al. [2019]). These difficulties can even be magnified when we have to deal with a multi-task environment. With these concerns, one promising approach is to perform offline reinforcement learning by aggregating information from multiple tasks. In this case, common information can be extracted from these tasks to improve the data efficiency.

While this approach has achieved tremendous success in a variety of applications Teh et al. [2017], Parisotto et al. [2015], Laroche and Barlier [2017], its theoretical understanding is still limited. A widely accepted assumption in the supervised learning literature is the existence of a common representation shared by different tasks. For example, Maurer et al. [2016] proposed a general method to learn data representation in multi-task supervised learning and learning-tolearn setting. Du et al. [2020] studied few-shot learning via representation learning with assumptions on a common representation among source and target tasks. Tripuraneni et al. [2021] focused on the problem

36th Conference on Neural Information Processing Systems (NeurIPS 2022).

of multi-task linear regression with low-rank representation, and proposed algorithms with sharp statistical rates. Moreover, Hu et al. [2021] studied the multi-task low-rank linear bandits problem and extend their method to a more general sequential decision learning setting.

Inspired by the theoretical results in multi-task online learning, we make the low rank assumption. For the multi-task offline linear bandits problem, where there are $M$ linear bandits with $d$-dimensional feature. The expected reward of arm $x_i \in \mathbb{R}^d$ for task $i$ is $\theta_i^\top x_i$, as determined by an unknown linear parameter $\theta_i$. To take advantage of the multi-task representation learning framework, we assume that $\theta_i$'s lie in an unknown $k$-dimensional subspace of $\mathbb{R}^d$, where $k$ is much smaller compared to $d$ and $M$ Yang et al. [2020]. The dependence among tasks make it possible to achieve smaller gap guarantee than to solve these tasks independently. Concretely speaking, the proposed offline actor-critic algorithm features a pessimism-inducing critic that utilizes a linear perturbation and an additional step of the low rank structure learning to improve value function approximation, while the actor updates the policy using a mirror descent algorithm that can be specialized as an exponentiated gradient updates method and advanced in a computational efficient style by considering the linear MDP function approximation and the soft-max policy class. Theorectically we prove the following sub-optimality gap for any policy and the learned mixture policy after $T$ episodes (see Theorem 1):

$$\begin{aligned}&\sum_{m=1}^{M} V_{1,m}^{\pi_m}(s_{1,m}) - V_{1,m}^{\pi_{\mathrm{ALG},m}}(s_{1,m}) \\ &\le \tilde{O}\left(\sqrt{Mk+kd}\sum_{h=1}^{H}\sqrt{\sum_{m=1}^{M}\left\{\mathbb{E}_{\pi_m}\|\phi(s,a)\|_{\Sigma_{h,m}^{-1}}\right\}^2} + MH\sqrt{\frac{\log|\mathcal{A}|}{T}}\right).\end{aligned}$$

This bound is significantly better than the naive bound if the coverage for every tasks is uniformly good (Section 4 Remark).

## 2 Background and formulation

We start our discussion by introducing some basic definitions and assumptions to formulate our problem setup.

### 2.1 Collection of markov decision processes

In this paper, we focus on a collections of finite-horizon Markov decision processes (MDPs for short) [Puterman, 2014, Bertsekas and Tsitsiklis, 1995, Sutton and Barto, 2018]. Suppose we have $M$ different MDPs $\{\mathcal{M}_m\}_{m=1}^{M}$ with a shared number of finite horizons $H$. The data were collected over each of the $H$ stages and the $M$ MDPs. We also assume the MDPs share a joint state space $\mathcal{S}$ (either discrete or continuous) and a discrete action space $\mathcal{A}$. For a given stage $h$ and a particular MDP $\mathcal{M}_m$, a reward function $r_{h,m}(s,a)$ is associated with the state action pairs $(s,a) \in \mathcal{S} \times \mathcal{A}$. Meanwhile, a matrix $\{\mathbb{P}_{h,m}(\cdot \mid s,a)\}_{(s,a)\in\mathcal{S}\times\mathcal{A}}$ is used to characterize the probability transition dynamics. When at horizon $h$, if the agent takes action $a$ under state $s$ in the $m$-th MDP, it receives a random reward drawn from a distribution $R_{h,m}(s,a)$ with mean $r_{h,m}(s,a)$, and then transitions randomly to a next state $s+$ from the transition probability $P_{h,m}(\cdot|s,a)$.

The agent adopts a policy at stage $h$ on the $m$-th MDP to pick an action $a$ to take based on some probability distribution $\pi_{h,m}(\cdot \mid s)$. Given a full policy $\pi_m = (\pi_{1,m}, \cdots, \pi_{H,m})$, the state-action value function over the $m$-th MDP at time step $h$ is given by

$$Q_{h,m}^{\pi_m}(s,a) = r_{h,m}(s,a) + \mathbb{E}\sum_{l=h+1}^{H} R_{l,m}(s_{l,m}, a_{l,m}),$$

where the expectation is taken over the trajectories induced by $\pi_m$ if starting from $(s,a)$. The value function follows naturally from the above definition:

$$V_{h,m}^{\pi_m}(s) = \langle \pi_{h,m}(\cdot|s), Q_{h,m}^{\pi_m}(s,\cdot)\rangle.$$

Under certain regularity conditions [Puterman, 2014, Shreve and Bertsekas, 1978], there always exists an optimal deterministic policy (under some regularity conditions) $\pi^\star$ for which it holds

$$V_{h,m}^{\pi^\star}(s) = \max_{\pi^m} V_{h,m}^{\pi_m}(s).$$

For convenience, we use the shorthand notations $V^{\star}_{h,m} = V^{\pi^\star}_{h,m}$ and $Q^{\star}_{h,m} = Q^{\pi^\star}_{h,m}$.

It is also useful to introduce the so called Bellman evaluation operator that is well studied in the context of Markov decision processes:

$$\mathcal{T}^{\pi}_{h,m}(Q_{h+1,m})(s,a) = r_{h,m}(s,a) + \mathbb{E}_{S'\sim\mathbb{P}_{h,m}(\cdot|s,a)}\mathbb{E}_{A'\sim\pi}Q_{h+1,m}(s',a').$$

Now we briefly discuss the data generating process for multiple MDPS. Suppose we collect $M$ dataset of the form

$$\mathcal{D}_m = \bigcup_{h=1}^{H}\mathcal{D}_{h,m} = \bigcup_{h=1}^{H}\{(s^i_{h,m}, a^i_{h,m}, r^i_{h,m}, s^{i,+}_{h,m})\}_{i=1}^{n_{h,m}}.$$

For convenience, we also denote the index set corresponding to $\mathcal{D}_{h,m}$ as $\mathcal{I}_{h,m}$.

The data are generated under the following assumption:

**Assumption 1** (Data generating assumptions)**.** *We assume that the state-action pairs are collected independently from some distribution $p(s,a)$ that is independent of horizon $h$ and MDP $\mathcal{M}_m$. Given (s,a), the rewards $r_{h,m}$ and next state $s^+$ was extracted from random distributions:*

$$r \sim R_{h,m}(s,a), s^+ \sim \mathbb{P}_{h,m}(\cdot \mid s,a).$$

*Moreover, we assume the reward distribution is sub-gaussian with parameter 1.*

### 2.2 Policy class and multi-task linear RL

In the framework of single-task linear RL, we assume that there exists a feature map $\phi : \mathcal{S}\times\mathcal{A}\to\mathbb{R}^d$ that maps each state-action pair to a d-dimensional vector in real space. This is useful especially when the dimension of the state space is continuous and has a large scale. In the multi-task linear MDP setting, we assume all the MDPs shared the same feature space. Using this feature representation, we further impose the linear MDP assumption; that is, at stage $h$, the state-value function of the $m$-th MDP is given by

$$Q^{\pi_m}_{h,m}(s,a) = \left\langle \phi(s,a)\,,\,\theta^{\pi_m}_{h,m}\right\rangle,\ m = 1,\cdots,M.$$

Generally speaking, this indicates that our analysis is built upon the linear function approximation class

$$\mathcal{Q}(\rho^\theta) = \{Q : (s,a) \to \left\langle \phi(s,a)\,,\,\theta\right\rangle \mid \|\theta\|_2 \le \rho^\theta, \rho^\theta \in (0,1]\}.$$

Also since we are working on the discrete action space, the soft-max policy class is of particular interest:

$$\Pi(\rho^\beta) = \left\{\frac{\exp(\left\langle \phi(s,a)\,,\,\beta\right\rangle)}{\sum_{a'}\exp(\left\langle \phi(s,a')\,,\,\beta\right\rangle)} \mid \|\beta\|_2 \le \rho^\beta\right\}.$$

However, it has been shown that such an assumption is not enough to guarantee a polynomial statistical complexity. We also need to require the $Q_h$ is approximately close under $\mathcal{T}_h$; that is, the inherent Bellman error is bounded as follows Zanette et al. [2021]:

**Assumption 2** (Bellman Restricted Closedness)**.** *The state-value function spaces $\mathcal{Q}$ are closed up to $\nu \in \mathbb{R}^H$ error in the sup-norm if there is a non-negative sequence $\{v_h\}$ such that for each $h \in [H]$, we have*

$$\max_{m\in[M]}\ \sup_{\substack{Q_{h+1,m}\in\mathcal{Q}_{h+1,m},\\ \pi_{h+1,m}\in\Pi_{h+1,m}}}\ \inf_{Q_{h,m}\in\mathcal{Q}_{m,h}} \|Q_{h,m} - \mathcal{T}^{\pi_{h+1,m}}_{m,h}Q_{h+1,m}\|_\infty \le \nu_h.$$

Due to Zanette et al. [2021], a sufficient assumption for Assumption 2 is given by the following low rank MDP condition, which gives $\nu_h = 0$:

**Assumption 3** (Low-rank MDP)**.** *An MDP is low-rank if for all $h \in [H]$, there exists a reward parameter $w_{h,m} \in \mathbb{R}^d$ and a component-wise positive mapping $\psi_h : \mathcal{S} \to \mathbb{R}^d_+$ such that $\|\psi_h(s)\|_1 = 1$ for all $s \in \mathcal{S}$ and*

$$r_{h,m} = \left\langle \phi(s,a)\,,\,w_{h,m}\right\rangle, \mathbb{P}_{h,m}(s' \mid s,a) = \left\langle \phi(s,a)\,,\,\psi_{h,m}(s')\right\rangle, \forall(s,a,h,s').$$

Besides these standard assumptions, we assume a joint low rank structure across all $M$ MDPs:

**Assumption 4** (Low-rank latent representation)**.** *If we aggregate all the latent parameters $\theta_{h,m}^{\pi_m}$ at stage $h$ into a matrix $\Theta_h = (\theta_{h,1}^{\pi_1}, \cdots, \theta_{h,M}^{\pi_M})$, there is an inherent low rank factorization that indicates a more parsimonious representation of the parameters:*

$$\Theta_h^\pi = B_h W_h^\pi = (B_h w_{h,1}^{\pi_1}, \cdots, B_h w_{h,M}^{\pi_M}).\ B_h \in \mathcal{O}^{d\times k},\ \textit{and}\ W_h^\pi \in \mathbb{R}^{k\times M}. \tag{1}$$

*Here $\mathcal{O}^{d\times k}$ contains matrices with orthonormal columns, and $w_{h,m}^{\pi_m}$ has bounded $\ell_2$ norm: $\|w_{h,m}^{\pi_m}\|_2 \le \rho^\theta$.*

Combining the original linear MDP assumption and the low rank representation assumption, another way to interpret such a low rank latent structure is that, using simple algebra we can show:

$$Q_{h,m}^{\pi_m}(s,a) = \langle \phi(s,a)\,,\,\theta_{h,m}^{\pi_m}\rangle = \langle B_h^\top \phi(s,a)\,,\,w_{h,m}^{\pi_m}\rangle.$$

This suggests that the inherent feature at the $h$-th stage lies in $\mathbb{R}^k$ instead of $\mathbb{R}^d$, which indicates the possibility of further reducing the algorithmic complexity by incorporating the structural information.

In this work we focus on bounding the following regret:

$$\text{Reg} = \sum_{m=1}^M \{V_{1,m}^\star(s_{1,m}) - V_{1,m}^{\pi_{\text{ALG},m}}(s_{1,m})\}$$

for some given starting state $s_{1,m}$ and learned policy $\pi_{\text{ALG},m}$.

## 3 Low rank actor critic algorithm

We are now ready to present an actor-critic algorithm to solve the multi-task offline policy learning problem.

### 3.1 Pessimistic Critic: low rank LSVI based global optimization

We begin with constructing a critic to generate pessimistic value function estimates corresponding to the sum $\sum_{m=1}^M V_{1,m}^{\pi_m}$. More specifically, we want to use the LSVI-based [Jin et al., 2020, Zanette et al., 2020] algorithms to build our (optimistic) estimator for the optimal value functions. Under the Bellman Restricted Closedness Assumption 2, it is reasonable to construct an estimator of $Q_{h,m}^{\pi_m}$ that at the end of episode $t-1$ by finding the best approximator in the function space $\mathcal{Q}_h$ of the Bellman transformation for the estimates of $Q_{h+1,m}^{\pi_m}$. The linear MDP assumption can further grant us a convenient approximation scheme by minimizing the least-square loss function. To extend the framework to a multi-task low rank LSVI setting, we simply propose to aggregate the losses across all $M$ tasks and try to minimize the joint least square approximation error.

To formalize the ideas in rigorous mathematics, given a collection of estimators $Q_{h+1,m}^{\pi_m}\left(\theta_{h+1,m}^{\pi_m^t}\right)$ for each $i \in [M]$ at the end of a certain episode (say episode $t$), we use the solution to the following constrained optimization problem

$$L(\Theta_h^t) = \sum_{m=1}^M \sum_{i\in\mathcal{D}_{h,m}} \left\{\phi(s_{h,m}^i, a_{h,m}^i)^\top \theta_{h,m}^t - r_{h,m}^i - V_{h+1,m}^t\left(\theta_{h+1,m}^t\right)\left(s_{h,m}^{i,+}\right)\right\}^2 \tag{2}$$

to approximate the Bellman update in the $t$-th episode. Recall that according to our previous definition,

$$\Theta_h^t = (\theta_{h,1}^t, \cdots, \theta_{h,M}^t)$$

is an aggregated parameter matrix at stage $h$ and episode $t$. As a small clarification, we are slightly abusing the notation here: the state-value function $Q_{h,m}^{\pi_m^t}$ and the parameter $\theta_{h,m}^{\pi_m^t}$ both depend on the policy obtained in the $t$-th episode; for brevity we suppress the policy dependence and mark the reliance on episode $t$ mainly through the $t$ in the superscript.

Now since the value function can be represented as a linear combination of the state-value function under a given policy, we have for any $s \in \mathcal{S}$,

$$V^t_{h+1,m}(\theta^t_{h+1,m})(s) = \sum_{a\in\mathcal{A}} \pi^t_{h+1,m}(a \mid s)\big\langle \phi(s,a)\,,\,\theta^t_{h+1,m}\big\rangle.$$

Plugging this equation into the definition of least square loss function (2), it is not hard to see that a straightforward update scheme for the parameter $\theta^t_{h,m}$ is given by

$$\begin{aligned}&\operatorname*{arg\,min}_{\|\theta^t_{h,m}\|_2\le D} L(\Theta^t_h)\\ &= \sum_{m=1}^{M}\sum_{i\in\mathcal{D}_{h,m}} \left\{\phi(s^i_{h,m}, a^i_{h,m})^\top\theta^t_{h,m} - r^i_{h,m} - \sum_{a\in\mathcal{A}}\pi^t_{h+1,m}(a \mid s^{i,+}_{h,m})\big\langle\phi(s^{i,+}_{h,m}, a)\,,\,\theta^t_{h+1,m}\big\rangle\right\}^2.\end{aligned}$$

However, this is not adequate in our setup since by Assumption 4, our parameter class is restricted to a low rank subspace. That being said, we need to incorporate the constraint

$$\Theta^t_h = B_h W^t_h.$$

This also motivates us to reparametrize the loss function using $B_h$ and $W^t_h$:

$$L(\Theta^t_h) := L(B_h, W^t_h). \tag{3}$$

Now we define the following cumulative covariance matrix of the features as well as two rescaled norms based on them:

$$\begin{aligned}\Sigma_{h,m}(\lambda) &= \sum_{i\in\mathcal{I}_{h,m}} \phi(s^i_{h,m}, a^i_{h,m})\phi(s^i_{h,m}, a^i_{h,m})^\top + \lambda I_d,\\ \|u\|^2_{\Sigma_{h,m}(\lambda)} &= u^\top\Sigma_{h,m}(\lambda)u,\ \|u\|^2_{\Sigma^{-1}_{h,m}(\lambda)} = u^\top\Sigma^{-1}_{h,m}(\lambda)u.\end{aligned}$$

To guarantee the pessimistic property of our estimator as well as utilize the low rank inherent representation, we extend the global optimization procedure of Zanette et al. [2021], Hu et al. [2021] which solves the following optimization problem in the $t$-th episode.

**Definition 1** (Global optimization procedure for the critic)**.** *Let $\{s_{1,m}\}_{m=1}^M$ be a set of starting states. At the $t$-th episode, we define a low rank global optimization procedure:*

$$\begin{aligned}\min_{\xi^t_{h,m},\widehat{\theta}^t_{h,m},\underline{\theta}^t_{h,m}} &\sum_{m=1}^{M}\sum_{a\in\mathcal{A}}\pi^t_{1,m}\left(a \mid s_{1,m}\right)\left\langle\phi\left(s_{1,i}, a\right), \underline{\theta}^t_{1,m}\right\rangle\\ \text{s.t. } &\left(\widehat{\theta}^t_{h,1},\ldots,\widehat{\theta}^t_{h,M}\right) = \widehat{B}^t_h(\widehat{w}^t_{h,1},\cdots,\widehat{w}^t_{h,M}) = \operatorname*{arg\,min}_{\left\|B^t_h w^t_{h,m}\right\|_2\le\rho^\theta} L\left(B_h, w^i_h\right);\\ &\underline{\theta}^t_{h,m} = \widehat{\theta}^t_{h,m} + \underline{\xi}^t_{h,m};\ \|\underline{\theta}^t_{h,m}\|_2 \le \rho^\theta;\\ &\sum_{m=1}^{M}\left\|\underline{\xi}^t_{h,m}\right\|^2_{\Sigma_{h,m}(\lambda)} \le (\alpha_h)^2.\end{aligned}$$

where the empirical least-square loss $L\left(B_h, w^i_h\right)$ was presented in (3).

We add some intuitive discussion on the optimization program 1. $\widehat{\theta}^t_{h,m}$ represents the solution of the low-rank least square minimization of the approximate value iteration. On one hand, it proceeds from stage $h+1$ to stage $h$ by finding the best Bellman transition approximation in the pre-specified linear function class. On the other hand, it imposes low rank constraints to reduce the scale of the parameter space and extract the shared information across different tasks. The global variables $\underline{\xi}^t_{h,m}$ are used to induce pessimism. Note that we do not add the bonus term directly on the approximated state-value function as in the tabular setting. Instead, to maintain the linearity of the program and avoid exponential propagation of errors, we manage to solve a robust optimization problem by adding these global optimization variables [Zanette et al., 2021, Hu et al., 2021].

Here we also want to make a brief comparison with the global optimization procedure introduced by Hu et al. [2021]. The main difference of their algorithm is that, since they consider the maximal policy in general, the objective function in their definition is given by

$$\sum_{m=1}^{M} \sum_{a\in\mathcal{A}} \max_{a\in\mathcal{A}} \left\langle \phi\left(s_{1,i}, a\right), \underline{\theta}^{t}_{1,m} \right\rangle,$$

which is a piece-wise linear function. Besides, since they carried out the analysis from an online setting, the optimization program focused on inducing optimism by maximizing the objective function.

Generally speaking, the global optimization procedure is not tractable in reality due to the complex dependence between stages as well as the low rank minimization step. One possible improvement is that: if we can accurately learn the $B_h$ matrix from some other data source and plug the estimator into the above optimization procedure, then the whole program can be simplified to a second order conic programming and become tractable.

**Algorithm 1:** Pessimistic critic

```
Input: Data set $\mathcal{D}$, low-rank parameter $k$, failure probability $\delta$, regularization $\lambda = 1$, inherent
       Bellman error $I$, current policy $\pi^t_m$.
Solve the global optimization problem 1.
Output: Optimum weight factor $\underline{\theta}^t_{h,m}$
```

**Algorithm 2:** Actor: Mirror descent

```
Input: Data set $\mathcal{D}$, starting state $s_1^m$, learning rate $\eta$.
Starting from uniform distributions by setting $\beta^1_{h,m} = (0, ..., 0)$.
for $t = 1, 2, ..., T$ do
    $Q^t_{h,m} = \texttt{CRITIC}(\mathcal{D}, \pi^t, s_1)$
    $\pi^{t+1}_{h,m}(a|s) \propto \pi^t_{h,m}(a|s)e^{\eta Q^t_{h,m}(s,a)}$
Output: Mixture policy $\{\pi^t_{h,m}\}_{t=1}^T$
```

### 3.2 The Actor: Mirror Descent

We now turn to the behavior of the actor. It applies the mirror descent algorithm based on the Kullback Leibler (KL) divergence [Bubeck et al., 2015]. It is well known that combining these two components leads to the exponentiated gradient update rule in every stage $h \in [H]$. Therefore, the soft-max policy in moving from iteration $t$ to $t+1$ can be updated as

$$\pi^{t+1}_{h,m}(a \mid s) \propto \pi^t_{h,m}(a \mid s)e^{\eta Q^t_{h,m}(s,a)} \quad \text{for each } (s,a) \in \mathcal{S} \times \mathcal{A}.$$

Here $\eta > 0$ is some learning rate. In the presentation of our theory we would specify a suitable stepsize to proceed. Moreover, the linearity assumption of the state-action function enables an even easier updating rule for the policies. More concretely speaking, if we have obtained

$$\pi^t_{h,m}(a \mid s) \propto e^{\eta \left\langle \phi(s,a)\,,\, \widehat{\beta}^t_{h,m} \right\rangle},$$

by running the critic, we know that the state-action function at the $t$-th episode is characterized by a pessimism parameter $\underline{\theta}^t_{h,m}$:

$$Q^t_{h,m}(s,a) = \left\langle \phi(s,a)\,,\, \underline{\theta}^t_{h,m} \right\rangle.$$

The mirror descent updating rule can then be simplified to

$$\begin{aligned} \pi^{t+1}_{h,m}(a|s) &\propto \pi^t_{h,m}(a|s)e^{\eta Q^t_{h,m}(s,a)} \\ &= e^{\eta \left\langle \phi(s,a)\,,\, \widehat{\beta}^t_{h,m} + \underline{\theta}^t_{h,m} \right\rangle}. \end{aligned}$$

Therefore, we can focus on updating the linear parameter as follows:

$$\widehat{\beta}^{t+1}_{h,m} = \widehat{\beta}^{t}_{h,m} + \underline{\theta}^{t}_{h,m}.$$

These calculations show that the update rule takes a much simpler and computationally efficient form for the actor. According to the comment of Zanette et al. [2021], in such cases, the function class and policy class $(\mathcal{Q}, \Pi)$ are said to be compatible [Sutton et al., 1999, Kakade, 2001, Raskutti and Mukherjee, 2015] and the resulting algorithm is often called the Natural Policy Gradient (NPG) (see also Geist et al. [2019], Shani et al. [2020]). This serves as the second benefit of inducing pessimism through a linear perturbation, since we can be guaranteed to have compatible function and policy classes thus a relatively easy linear updating rule for the actor.

## 4 Main result

We now provide a guarantee of the performance of the policy output by our algorithm in the Low-rank MDP setting. The upper bound of the gap depends on two terms. The first term is the *uncertainty error*, which characterize the uncertainty of the optimal policy given the data set $\mathcal{D}_m$. The information of the data set is completely characterized by the uncertainty error. Suppose the accumulate covariance matrix $\Sigma_h$ cover the policy $\pi_m$ well, then the term $\mathbb{E}_{\pi_m}\|\phi(s,a)\|_{\Sigma^{-1}_{h,m}}$ is small. The second term is the *optimization error* incurred by Mirror Descent which scales as $O(HM\sqrt{\frac{\log|\mathcal{A}|}{T}})$, this term decays as the number of iteration increase.

**Theorem 1** (Gap Guarantee for Low-rank MDP). *Suppose we are given a data set $\mathcal{D}$ collected in the way that follows Assumption 1, then we set $\alpha_h = \tilde{O}(\sqrt{Mk + kd})$. After running $T \geq \log|\mathcal{A}|$ rounds of iteration with stepsize $\eta = \sqrt{\frac{\log|\mathcal{A}|}{T}}$, the algorithm returns $M$ mixture policies $\pi_{\mathrm{ALG},m}$ that satisfies,*

$$\begin{aligned}&\sum_{m=1}^{M} V^{\pi_m}_{1,m}(s_{1,m}) - V^{\pi_{\mathrm{ALG},m}}_{1,m}(s_{1,m})\\ \leq &\tilde{O}\left(\sqrt{Mk+kd}\sum_{h=1}^{H}\sqrt{\sum_{m=1}^{M}(\mathbb{E}_{\pi_m}\|\phi(s,a)\|_{\Sigma^{-1}_{h,m}})^2} + MH\sqrt{\frac{\log|\mathcal{A}|}{T}}\right).\end{aligned}$$

*for any policy $\pi_m$ uniformly with probability at least $1-\delta$.*

**Remark:** If we solve the $M$ tasks independently without sharing the information, then we can only bound the gap by $\tilde{O}\left(\sqrt{d}\sum_{h=1}^{H}\sum_{m=1}^{M}\mathbb{E}_{\pi_m}\|\phi(s,a)\|_{\Sigma^{-1}_{h,m}} + MH\sqrt{\frac{\log|\mathcal{A}|}{T}}\right)$. We note that when the coverage for every tasks are uniformly good, *i.e.*, for some small constant $C$,

$$\sqrt{M}\sqrt{\sum_{m=1}^{M}(\mathbb{E}_{\pi_m}\|\phi(s,a)\|_{\Sigma^{-1}_{h,m}})^2} \leq C\sum_{m=1}^{M}\mathbb{E}_{\pi_m}\|\phi(s,a)\|_{\Sigma^{-1}_{h,m}}$$

Note that if for every tasks, $\mathbb{E}_{\pi_m}\|\phi(s,a)\|_{\Sigma^{-1}_{h,m}}$ takes the same value, then $C = 1$. Follow by this assumption, the bound given by the theorem is

$$\tilde{O}\left(C\sqrt{k + kd/M}\sum_{h=1}^{H}\sum_{m=1}^{M}\mathbb{E}_{\pi_m}\|\phi(s,a)\|_{\Sigma^{-1}_{h,m}} + MH\sqrt{\frac{\log|\mathcal{A}|}{T}}\right)$$

The term $O(\sqrt{k + kd/M})$ is significantly smaller than $O(\sqrt{d})$ if $k \ll d$ and $k \ll M$. However, if the coverage is not uniform, our bound might be larger than the naive bound, which we believe is due to our coarse analysis.

## 5 Conclusion and Discussion

In this work we utilize the low dimensional dependence structure among a collection of offline reinforcement learning tasks to achieve smaller gap guarantee. The proposed critic induces pessimism utilizing a linear perturbation and cultivates low dimension structure through an additional step of the low rank learning, while the actor updates the policy using a mirror descent algorithm that can be specialized as an exponentiated gradient updates method and advanced in a computational efficient style. Theoretically, we prove the sub-optimality gap for any policy and the learned mixture policy after $T$ episodes. When the coverage for every task is uniformly good and the inherent dimension $k$ is far smaller than the dimension of the feature space and the number of running tasks, the sub-optimality gap is significantly smaller than that given by the naive approach, which validates the advantage of representation learning in multi-task RL settings.

Although our work takes a step forward in bridging the gap between the statistical complexity and algorithm implementation using an actor-critic framework, the low rank feature learning step in the global optimization is still hard to solve in practice. Therefore, the first question of interest is how to further improve the algorithms to simultaneously allow solvable low rank feature extraction and policy learning. If this can be achieved, the next interesting question would be evaluating the performance of this pessimism based algorithm in some simulation settings and real-life scenarios and understand its practical behavior. Another question to ask is that, our results show that when the number of target MDP is large, we could benefit significantly from representation learning. If we only hope to transfer knowledge from many MDPs to one target MDP, how should we modify the algorithm or the analysis and argue that such benefits still exist (or at least prove incorporating knowledge from relevant MDPs wouldn't hurt statistical efficiency)? This question is related to transferred learning that has received wide attention these days. Last but not least, in some cases, we believe that linear MDP is a relatively restrictive and specific setup. It is worthy of further exploration to extend these ideas to more general function approximation schemes.

# References

Jens Kober, J Andrew Bagnell, and Jan Peters. Reinforcement learning in robotics: A survey. *The International Journal of Robotics Research*, 32(11):1238–1274, 2013.

Jonathan Bassen, Bharathan Balaji, Michael Schaarschmidt, Candace Thille, Jay Painter, Dawn Zimmaro, Alex Games, Ethan Fast, and John C Mitchell. Reinforcement learning for the adaptive scheduling of educational activities. In *Proceedings of the 2020 CHI Conference on Human Factors in Computing Systems*, pages 1–12, 2020.

Ahmad EL Sallab, Mohammed Abdou, Etienne Perot, and Senthil Yogamani. Deep reinforcement learning framework for autonomous driving. *Electronic Imaging*, 2017(19):70–76, 2017.

Omer Gottesman, Fredrik Johansson, Matthieu Komorowski, Aldo Faisal, David Sontag, Finale Doshi-Velez, and Leo Anthony Celi. Guidelines for reinforcement learning in healthcare. *Nature medicine*, 25(1):16–18, 2019.

Yee Teh, Victor Bapst, Wojciech M Czarnecki, John Quan, James Kirkpatrick, Raia Hadsell, Nicolas Heess, and Razvan Pascanu. Distral: Robust multitask reinforcement learning. *Advances in neural information processing systems*, 30, 2017.

Emilio Parisotto, Jimmy Lei Ba, and Ruslan Salakhutdinov. Actor-mimic: Deep multitask and transfer reinforcement learning. *arXiv preprint arXiv:1511.06342*, 2015.

Romain Laroche and Merwan Barlier. Transfer reinforcement learning with shared dynamics. In *Thirty-First AAAI Conference on Artificial Intelligence*, 2017.

Markus Maurer, J Christian Gerdes, Barbara Lenz, and Hermann Winner. *Autonomous driving: technical, legal and social aspects*. Springer Nature, 2016.

Simon S Du, Wei Hu, Sham M Kakade, Jason D Lee, and Qi Lei. Few-shot learning via learning the representation, provably. *arXiv preprint arXiv:2002.09434*, 2020.

Nilesh Tripuraneni, Chi Jin, and Michael Jordan. Provable meta-learning of linear representations. In *International Conference on Machine Learning*, pages 10434–10443. PMLR, 2021.

Jiachen Hu, Xiaoyu Chen, Chi Jin, Lihong Li, and Liwei Wang. Near-optimal representation learning for linear bandits and linear rl. In *International Conference on Machine Learning*, pages 4349–4358. PMLR, 2021.

Jiaqi Yang, Wei Hu, Jason D Lee, and Simon S Du. Impact of representation learning in linear bandits. *arXiv preprint arXiv:2010.06531*, 2020.

Martin L Puterman. *Markov decision processes: discrete stochastic dynamic programming*. John Wiley & Sons, 2014.

Dimitri P Bertsekas and John N Tsitsiklis. Neuro-dynamic programming: an overview. In *Proceedings of 1995 34th IEEE conference on decision and control*, volume 1, pages 560–564. IEEE, 1995.

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

Steven E Shreve and Dimitri P Bertsekas. Alternative theoretical frameworks for finite horizon discrete-time stochastic optimal control. *SIAM Journal on control and optimization*, 16(6):953–978, 1978.

Andrea Zanette, Martin J Wainwright, and Emma Brunskill. Provable benefits of actor-critic methods for offline reinforcement learning. *Advances in neural information processing systems*, 34, 2021.

Chi Jin, Zhuoran Yang, Zhaoran Wang, and Michael I Jordan. Provably efficient reinforcement learning with linear function approximation. In *Conference on Learning Theory*, pages 2137–2143. PMLR, 2020.

Andrea Zanette, Alessandro Lazaric, Mykel Kochenderfer, and Emma Brunskill. Learning near optimal policies with low inherent bellman error. In *International Conference on Machine Learning*, pages 10978–10989. PMLR, 2020.

Sébastien Bubeck et al. Convex optimization: Algorithms and complexity. *Foundations and Trends® in Machine Learning*, 8(3-4):231–357, 2015.

Richard S Sutton, David McAllester, Satinder Singh, and Yishay Mansour. Policy gradient methods for reinforcement learning with function approximation. *Advances in neural information processing systems*, 12, 1999.

Sham M Kakade. A natural policy gradient. *Advances in neural information processing systems*, 14, 2001.

Garvesh Raskutti and Sayan Mukherjee. The information geometry of mirror descent. *IEEE Transactions on Information Theory*, 61(3):1451–1457, 2015.

Matthieu Geist, Bruno Scherrer, and Olivier Pietquin. A theory of regularized markov decision processes. In *International Conference on Machine Learning*, pages 2160–2169. PMLR, 2019.

Lior Shani, Yonathan Efroni, and Shie Mannor. Adaptive trust region policy optimization: Global convergence and faster rates for regularized mdps. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 5668–5675, 2020.