Efficient Reinforcement Learning via Representation Learning

Wen Sun

Joint work with Masatoshi Uehara (Cornell) & Xuezhou Zhang (Princeton)
Empirical RL for large-scale problems

Rich (nonlinear) function approximation + RL can work well with enough samples
Can we design provably efficient algorithms for

*Rich Function Approx + RL*?
Can we design provably efficient algorithms for

Rich Function Approx + RL?

Dataset $\mathcal{D}$

Environment w/ complex high-dim data
Can we design provably efficient algorithms for

*Rich Function Approx + RL*?

**Our solution:**

**Representation Learning Oracle:**

\[ \phi(s, a) \in \mathbb{R}^d \]

**Dataset** \( \mathcal{D} \)

**Environment with complex high-dim data**

**RL using \( \phi \)**
Episodic Infinite Horizon Discounted MDPs

Policy: state to action
\[ \pi(s) \rightarrow a \]

Reward & Next State
\[ r(s, a), s' \sim P(\cdot \mid s, a) \]
Episodic Infinite Horizon Discounted MDPs

Policy: state to action

\[ \pi(s) \rightarrow a \]

Reward & Next State

\[ r(s, a), s' \sim P(\cdot|s, a) \]

Objective:

\[
\max_{\pi} J(\pi; P, r), \text{ where } J(\pi; P, r) := \mathbb{E} \left[ r(s_0, a_0) + \gamma r(s_1, a_1) + \gamma^2 r(s_2, a_2) + \ldots | a \sim \pi, P \right]
\]
Episodic Infinite Horizon Discounted MDPs

Policy: state to action

\[ \pi(s) \rightarrow a \]

Reward & Next State

\[ r(s, a), s' \sim P( \cdot | s, a) \]

Objective:

\[
\max_{\pi} J(\pi; P, r), \text{ where } J(\pi; P, r) := \mathbb{E} \left[ r(s_0, a_0) + \gamma r(s_1, a_1) + \gamma^2 r(s_2, a_2) + \ldots | a \sim \pi, P \right]
\]

Assume fixed initial state \( s_0 \)
Low-rank MDP

Transition matrix $P \in \mathbb{R}^{S_A \times S}$ has rank $d$
Low-rank MDP

Transition matrix $P \in \mathbb{R}^{SA \times S}$ has rank $d$

$$\exists \mu^*, \phi^* : \forall s, a, s', P^*(s'|s, a) = \mu^*(s')^\top \phi^*(s, a)$$
Low-rank MDP

Transition matrix $P \in \mathbb{R}^{SA \times S}$ has rank $d$

$$\exists \mu^*, \phi^* : \forall s, a, s', P^*(s'|s, a) = \mu^*(s')^\top \phi^*(s, a)$$

Low-rank MDP $\neq$ Linear MDPs (Jin et al, Yang & Wang)

Linear MDP = low-rank + known $\phi^*$
Low-rank MDP

Transition matrix $P \in \mathbb{R}^{S \times A \times S}$ has rank $d$

\[ \exists \mu^*, \phi^*: \forall s, a, s', P^*(s' | s, a) = \mu^*(s')^\top \phi^*(s, a) \]

Low-rank MDP $\neq$ Linear MDPs (Jin et al, Yang & Wang)

Linear MDP = low-rank + known $\phi^*$
Low-rank MDP is general

e.g., Latent variable models where $Z$ is the discrete latent space

$\phi^*(s, a) \in \Delta(Z)$
Low-rank MDP is general

e.g., Latent variable models where $Z$ is the discrete latent space

$\phi^*(s, a) \in \Delta(Z)$

Given $s, a$: $z \sim \phi^*(s, a), s' \sim \nu^*(z)$
Provably efficient learning in low-rank mdp is plausible

<table>
<thead>
<tr>
<th>Setting</th>
<th>Sample Complexity</th>
<th>Computation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Olive [JKALS,17]</td>
<td>$\frac{d^2 A}{e^2(1 - \gamma)^4}$</td>
<td>Inefficient</td>
</tr>
<tr>
<td>Witness Rank [SJKAL,19]</td>
<td>$\frac{d^2 A}{e^2(1 - \gamma)^4}$</td>
<td>Inefficient</td>
</tr>
<tr>
<td>BLin-UCB [DKLLMSW,21]</td>
<td>$\frac{d^2 A}{e^2(1 - \gamma)^7}$</td>
<td>Inefficient</td>
</tr>
<tr>
<td>Moffle [MCKJA,21]</td>
<td>$\frac{d^6 A^{13}}{e^2\eta^5(1 - \gamma)^5}$</td>
<td>Oracle-efficient</td>
</tr>
<tr>
<td>FLAMBE [AKKS,20]</td>
<td>$\frac{d^7 A^9}{e^{10}(1 - \gamma)^{22}}$</td>
<td>Oracle-efficient</td>
</tr>
</tbody>
</table>
Provably efficient learning in low-rank MDP is plausible

<table>
<thead>
<tr>
<th>Setting</th>
<th>Sample Complexity</th>
<th>Computation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Olive [JKALS,17]</td>
<td>( \frac{d^2 A}{\epsilon^2(1 - \gamma)^4} )</td>
<td>Inefficient</td>
</tr>
<tr>
<td>Witness Rank [SJKAL,19]</td>
<td>( \frac{d^2 A}{\epsilon^2(1 - \gamma)^4} )</td>
<td>Inefficient</td>
</tr>
<tr>
<td>BLin-UCB [DKLLMSW,21]</td>
<td>( \frac{d^2 A}{\epsilon^2(1 - \gamma)^7} )</td>
<td>Inefficient</td>
</tr>
<tr>
<td>Moffle [MCKJA,21]</td>
<td>( \frac{d^6 A^{13}}{\epsilon^2 \eta^5 (1 - \gamma)^5} )</td>
<td>Oracle-efficient</td>
</tr>
<tr>
<td>FLAMBE [AKKS,20]</td>
<td>( \frac{d^7 A^9}{\epsilon^{10} (1 - \gamma)^{22}} )</td>
<td>Oracle-efficient</td>
</tr>
</tbody>
</table>

FLAMBE is oracle-efficient and was state-of-art on low-rank MDP
Our learning setting

1. Realizable hypothesis classes $\Gamma, \Phi$

$$\mu^* \in \Gamma, \phi^* \in \Phi$$
Our learning setting

1. Realizable hypothesis classes $\Gamma, \Phi$

$$\mu^* \in \Gamma, \phi^* \in \Phi$$

2. Computation oracle:

Maximum Likelihood Estimation (MLE):

$$(\hat{\mu}, \hat{\phi}) := \arg \max_{\mu, \phi} \sum_{i=1}^{n} \ln \left( \mu(s_i^\prime) \phi(s_i, a_i) \right)$$
Our learning setting

1. Realizable hypothesis classes $\Gamma, \Phi$

   \[ \mu^* \in \Gamma, \phi^* \in \Phi \]

2. Computation oracle:

   Maximum Likelihood Estimation (MLE):

   \[ (\hat{\mu}, \hat{\phi}) := \arg \max_{\mu,\phi} \sum_{i=1}^n \ln (\mu(s_i')\phi(s_i, a_i)) \]

3. Learning Goal:

   Finding near-optimal policy w/ (tight) $\text{poly}(A, d, 1/(1 - \gamma), \ln(|\Phi| |\Gamma|))$
Our algorithm: Rep-UCB
(UCB-driven Representation Learning for online RL)

At iteration n:
Our algorithm: Rep-UCB
(UCB-driven Representation Learning for online RL)

At iteration $n$:

Data generation from $\pi^n$:

$s \sim d^n$, $a \sim \text{Uniform}(\mathcal{A})$, $s' \sim P^*(\cdot | s, a)$

$a' \sim \text{Uniform}(\mathcal{A})$, $s'' \sim P^*(\cdot | s', a')$
Our algorithm: Rep-UCB

(UCB-driven Representation Learning for online RL)

At iteration $n$:

Data generation from $\pi^n$:

$s \sim \mathcal{d}^\pi, a \sim \text{Uniform}(\mathcal{A}), s' \sim P^*(\cdot | s, a)$

$a' \sim \text{Uniform}(\mathcal{A}), s'' \sim P^*(\cdot | s', a')$

Data Aggregation:

$\mathcal{D}_n = \mathcal{D}_{n-1} + \{s, a, s'\}$

$\mathcal{D}'_n = \mathcal{D}'_{n-1} + \{s', a', s''\}$
Our algorithm: Rep-UCB

(UCB-driven Representation Learning for online RL)

At iteration n:

Data generation from $\pi^n$:

$s \sim d^n, a \sim \text{Uniform}(\mathcal{A}), s' \sim P^*(. | s, a)$

$a' \sim \text{Uniform}(\mathcal{A}), s'' \sim P^*(. | s', a')$

Data Aggregation:

$\mathcal{D}_n = \mathcal{D}_{n-1} + \{s, a, s'\}$

$\mathcal{D}'_n = \mathcal{D}'_{n-1} + \{s', a', s''\}$

Representation / model Learning (MLE):

$\hat{P} := (\hat{\mu}, \hat{\phi}) = \arg \max_{\mu, \phi} \mathbb{E}_{\mathcal{D}_n + \mathcal{D}'_n} \ln(\mu(s')^T \phi(s, a))$
Our algorithm: Rep-UCB  
(UCB-driven Representation Learning for online RL)

At iteration n:

Data generation from $\pi^n$:

$s \sim d^{\pi^n}, a \sim \text{Uniform}(\mathcal{A}), s' \sim P^*(\cdot | s, a)$
$a' \sim \text{Uniform}(\mathcal{A}), s'' \sim P^*(\cdot | s', a')$

Data Aggregation:

$\mathcal{D}_n = \mathcal{D}_{n-1} + \{s, a, s'\}$
$\mathcal{D}_n' = \mathcal{D}_{n-1}' + \{s', a', s''\}$

(Linear bandit style) bonus under $\hat{\phi}$:

$b(s, a) = c\sqrt{\hat{\phi}(s, a)\Sigma^{-1}\hat{\phi}(s, a)}$

$\Sigma = \sum_{s,a \in \mathcal{D}_n} \hat{\phi}(s, a)\hat{\phi}(s, a)^T$

Representation / model Learning (MLE)

$\hat{P} := (\hat{\mu}, \hat{\phi}) = \arg\max_{\mu,\phi} \mathbb{E}_{\mathcal{D}_n+\mathcal{D}_n'} \ln(\mu(s')^T\phi(s, a))$
Our algorithm: Rep-UCB
(UCB-driven Representation Learning for online RL)

At iteration n:

Data generation from $\pi^n$:
\[
s \sim d^n, a \sim \text{Uniform}(\mathcal{A}), s' \sim P^*(\cdot | s, a) \\
a' \sim \text{Uniform}(\mathcal{A}), s'' \sim P^*(\cdot | s', a')
\]

Plan w/ reward + bonus
\[
\pi^{n+1} = \max_{\pi} J(\pi; \hat{P}, r + b)
\]

(Linear bandit style) bonus under $\hat{\phi}$:
\[
b(s, a) = c \sqrt{\hat{\phi}(s, a) \Sigma^{-1} \hat{\phi}(s, a)}
\]
\[
\Sigma = \sum_{s, a \in \mathcal{D}_n} \hat{\phi}(s, a)\hat{\phi}(s, a)^T
\]

Data Aggregation:
\[
\mathcal{D}_n = \mathcal{D}_{n-1} + \{s, a, s'\} \\
\mathcal{D}'_n = \mathcal{D}'_{n-1} + \{s', a', s''\}
\]

Representation / model Learning (MLE)
\[
\hat{P} := (\hat{\mu}, \hat{\phi}) = \arg \max_{\mu, \phi} \mathbb{E}_{\mathcal{D}_n + \mathcal{D}'_n} \ln(\mu(s')^T \phi(s, a))
\]
PAC-Bound of Rep-UCB in low-rank MDP

Assume trajectory-reward is normalized in [0,1]. With high probability, it finds an $\epsilon$ near optimal policy, with # of samples:

$$\tilde{O}\left(\frac{d^4A^2}{\epsilon^2(1-\gamma)^5} \cdot \ln(\frac{|\Gamma||\Phi|}{\epsilon})\right)$$
PAC-Bound of Rep-UCB in low-rank MDP

Assume trajectory-reward is normalized in [0,1]. With high probability, it finds an $\epsilon$ near optimal policy, with # of samples:

$$\tilde{O}\left(\frac{d^4A^2}{\epsilon^2(1-\gamma)^5} \cdot \ln\left( \frac{|\Gamma||\Phi|}{5} \right)\right)$$

For reference, prior SOTA FLAMBE has the following bound:

$$\tilde{O}\left(\frac{d^7A^9}{\epsilon^5(1-\gamma)^{22}} \cdot \ln\left( \frac{|\Gamma||\Phi|}{5} \right)\right)$$
Applying our new techniques to Offline RL

Offline RL: we only have a **static** dataset $\mathcal{D} = \{s, a, s'\}$, where

$$(s, a) \sim \pi_b, s' \sim P^*(. | s, a)$$
Applying our new techniques to Offline RL

Offline RL: we only have a static dataset $\mathcal{D} = \{s, a, s'\}$, where $(s, a) \sim \pi_b$, $s' \sim P^*(\cdot | s, a)$

Goal: learn to find some high quality policy solely from $\mathcal{D}$

[Image from BAIR blog post: https://bair.berkeley.edu/blog/2020/12/07/offline/]
Coverage condition of the offline data

A comparator policy $\pi$ is covered by offline data if the relative condition number is bounded:

$$C_{\pi^*} := \max_x \frac{x^T \left( \mathbb{E}_{s,a \sim d^\pi} \phi^*(s, a) \phi^*(s, a)^T \right) x}{x^T \left( \mathbb{E}_{s,a \sim d^\pi_b} \phi^*(s, a) \phi^*(s, a)^T \right) x} < \infty$$

Note coverage is wrt true representation only!
Coverage condition of the offline data

A comparator policy $\pi$ is covered by offline data if the relative condition number is bounded:

$$C_{\pi^*} := \max_x \frac{x^T \left( \mathbb{E}_{s,a \sim d\pi} \phi^*(s, a) \phi^*(s, a)^T \right) x}{x^T \left( \mathbb{E}_{s,a \sim d\pi_b} \phi^*(s, a) \phi^*(s, a)^T \right) x} < \infty$$

Note coverage is wrt true representation only!

Goal is to learn robustly, i.e., as long as there is a high quality policy that is covered by $d^{\pi_b}$, we want to compete against it!
The Rep-LCB Algorithm
(Low confidence bound driven offline RL)

1. Representation / model Learning (MLE) under $\mathcal{D}$

$\hat{P} := (\hat{\mu}, \hat{\phi}) = \arg \max_{\mu, \phi} \mathbb{E}_{\mathcal{D}} \ln(\mu(s')^T \phi(s, a))$
The Rep-LCB Algorithm
(Low confidence bound driven offline RL)

1. Representation / model Learning (MLE) under $\hat{D}$:
$$\hat{P} := (\hat{\mu}, \hat{\phi}) = \arg \max_{\mu, \phi} \mathbb{E}_{\hat{D}} \ln(\mu(s')^\top \phi(s))$$

2. Penalty w/ $\hat{\phi}$:
$$b(s, a) = c \sqrt{\hat{\phi}(s, a) \Sigma^{-1} \hat{\phi}(s, a)}$$
$$\Sigma = \sum_{s, a \in \hat{D}_n} \hat{\phi}(s, a) \hat{\phi}(s, a)^\top$$
The Rep-LCB Algorithm
(Low confidence bound driven offline RL)

1. Representation / model Learning (MLE) under $\mathcal{D}$:

$$\hat{\mu}, \hat{\phi} = \arg\max_{\mu, \phi} \mathbb{E}_{\mathcal{D}} \ln(\mu(s')^T \phi(s))$$

2. Penalty w/ $\hat{\phi}$

$$b(s, a) = c \sqrt{\hat{\phi}(s, a) \Sigma^{-1} \hat{\phi}(s, a)}$$

$$\Sigma = \sum_{s, a \in \mathcal{D}_n} \hat{\phi}(s, a) \hat{\phi}(s, a)^T$$

3. Conservative Plan w/ reward + penalty

$$\hat{\pi} = \max_{\pi} J(\pi; \hat{P}, r - b)$$
The guarantee of Rep-LCB

Assume the behavior policy $\pi_b(a \mid s) \geq w, \forall s$; W/ high probability, for ALL comparator policy $\pi^*$ (include history-dependent ones):

$$J(\pi^*; r) - J(\hat{\pi}; r) \leq \tilde{O} \left( \frac{d^2}{(1 - \gamma)^{1.5}} \sqrt{\frac{wC_{\pi^*}}{n}} \cdot \ln(|\Phi| ||\Gamma||) \right)$$
The guarantee of Rep-LCB

Assume the behavior policy \( \pi_b(a \mid s) \geq w, \forall s \); W/ high probability, for ALL comparator policy \( \pi^* \) (include history-dependent ones):

\[
J(\pi^*; r) - J(\hat{\pi}; r) \leq \widetilde{O} \left( \frac{d^2}{(1 - \gamma)^{1.5}} \sqrt{\frac{wC_{\pi^*}}{n}} \cdot \ln(\Phi \Gamma) \right)
\]
The guarantee of Rep-LCB

Assume the behavior policy \( \pi_b(a \mid s) \geq w, \forall s; \) W/ high probability, for **ALL** comparator policy \( \pi^* \) (include **history-dependent** ones):

\[
J(\pi^*; r) - J(\hat{\pi}; r) \leq \tilde{O}\left(\frac{d^2}{(1-\gamma)^{1.5}} \sqrt{\frac{wC_{\pi^*}}{n}} \cdot \ln(|\Phi| ||\Gamma||)\right)
\]

(prior work CPPO [Uehara & Sun, 21] can achieve similar guarantee, but is a version-space alg)
Summary

1. Improved online Representation Learning algorithm for low-rank MDP:
   Oracle-efficient + tight sample complexity

2. New offline RL algorithm for low-rank MDP:
   Partial coverage + Oracle-efficient