Sample-Efficient Exploration in RL with Function Approximation

Nan Jiang

Joint with: Akshay Krishnamurthy, Alekh Agarwal, John Langford, Rob Schapire
3 core challenges of RL

- Long-term planning
- Generalization
- Exploration
3 core challenges of RL

✓ Long-term planning

✓ Generalization

Exploreation X
3 core challenges of RL

✓ Long-term planning

Approximate DP

X Generalization

PAC-MDP

Exploration ✓
3 core challenges of RL

- Long-term planning
- Generalization
- Exploration

Approximate DP

PAC-MDP
3 core challenges of RL

- Long-term planning
- Approximate DP
- Generalization (Supervised Learning)
- Exploration (Multi-Armed Bandit)
- (Dynamic Programming)
- PAC-MDP
3 core challenges of RL

Bellman equation
(Dynamic Programming)
Long-term planning

Approximate DP

Generalization
(Supervised Learning)
Statistical complexity
(e.g., VC-dimension)

Exploration
(Multi-Armed Bandit)
Optimism in face of uncertainty

PAC-MDP
3 core challenges of RL

- Bellman equation (Dynamic Programming)
- Long-term planning

Our Contributions
- Measure: Bellman rank
- Algorithm: OLIVE

Generalization
(Supervised Learning)
- Statistical complexity (e.g., VC-dimension)

Approximate DP

Exploration
(Multi-Armed Bandit)
- Optimism in face of uncertainty

PAC-MDP
Random exploration can be inefficient
Random exploration can be inefficient
Random exploration can be inefficient
Random exploration can be inefficient

visited in $2^{-H}$ fraction of all trajectories
Random exploration can be inefficient

visited in $2^{-H}$ fraction of all trajectories

Freeway (one of the Atari games)

“Freeway + RL”: https://youtu.be/44CilPmlimQ
Random exploration can be inefficient

visited in $2^{-H}$ fraction of all trajectories

Freeway (one of the Atari games)

“Freeway + RL”: https://youtu.be/44CilPmlimQ
List of games

At human-level or above

Below human-level

[Mnih et al’15]
Exploration
• Learner gathers own data

List of games

At human-level or above
Below human-level

“hard exploration”  [Bellemare et al’16]
“tabular RL”

Exploration in small state space is tractable
Exploration in small state space is tractable
• Optimize chances for reaching under-visited states
Exploration in small state space is tractable

- Optimize chances for reaching under-visited states
- Sample complexity = $\text{poly}(|X|)$ (and $|A|$, $H$, $1/\varepsilon$, $1/\delta$)

“PAC-MDP” [Kearns & Singh’98] [Brafman & Tennenholtz’02] …
Exploration in small state space is tractable

- Optimize chances for reaching under-visited states
- Sample complexity $= \text{poly}(|X|)$ (and $|A|, H, 1/\epsilon, 1/\delta$)

“PAC-MDP” [Kearns & Singh’98] [Brafman & Tennenholtz’02] …

Generalization

- Large state space

“tabular RL”
Exploration
• Learner gathers own data

Generalization
• Large state space

Systematic exploration in large state spaces, at least information-theoretically?
Formal Model

• Episodic MDP with horizon $H$

• In each episode: for $h = 1, \ldots, H$, learner
Formal Model

• Episodic MDP with horizon $H$

• In each episode: for $h = 1, \ldots, H$, learner
  • observes state feature $x_h \in X$ (possibly infinite) (w.l.o.g. $x_1 = x^0$)
Formal Model

• Episodic MDP with horizon $H$

• In each episode: for $h = 1, \ldots, H$, learner
  • observes state feature $x_h \in X$ (possibly infinite) (w.l.o.g. $x_1 = x^0$)
  • chooses action $a_h \in A$ (finite & manageable)
Formal Model

• Episodic MDP with horizon $H$

• In each episode: for $h = 1, \ldots, H$, learner
  • observes state feature $x_h \in X$ (possibly infinite) (w.l.o.g. $x_1 = x^0$)
  • chooses action $a_h \in A$ (finite & manageable)
  • receives reward $r_h \in \mathbb{R}$ (bounded)
Formal Model

- Episodic MDP with horizon $H$
- In each episode: for $h = 1, \ldots, H$, learner
  - observes state feature $x_h \in X$ (possibly infinite) (w.l.o.g. $x_1 = x^0$)
  - chooses action $a_h \in A$ (finite & manageable)
  - receives reward $r_h \in R$ (bounded)
- Learning goal: given $F$ such that $Q^* \in F$, (will relax)

\[ F = \{ f(\cdot; \theta) : \theta \in \Theta \} \]
Formal Model

- Episodic MDP with horizon $H$

- In each episode: for $h = 1, \ldots, H$, learner
  - observes state feature $x_h \in X$ (possibly infinite) (w.l.o.g. $x_1 = x^0$)
  - chooses action $a_h \in A$ (finite & manageable)
  - receives reward $r_h \in \mathbb{R}$ (bounded)

- Learning goal: given $F$ such that $Q^* \in F$, (will relax)
  w.p. $1 - \delta$, find policy $\pi$ s.t. $\nu^* - \nu^\pi \leq \varepsilon$

\[ \mathcal{F} = \{ f(\cdot; \theta) : \theta \in \Theta \} \]
Formal Model

- Episodic MDP with horizon $H$
- In each episode: for $h = 1, \ldots, H$, learner
  - observes state feature $x_h \in X$ (possibly infinite) (w.l.o.g. $x_1 = x^0$)
  - chooses action $a_h \in A$ (finite & manageable)
  - receives reward $r_h \in R$ (bounded)
- Learning goal: given $F$ such that $Q^* \in F$, (will relax)
w.p. $1 - \delta$, find policy $\pi$ s.t. $v^* - v^\pi \leq \epsilon$
using $\text{poly}(|A|, H, \log|F|, 1/\epsilon, 1/\delta)$ episodes. (can extend to VC-dim)

$$\mathcal{F} = \{f(\cdot; \theta) : \theta \in \Theta\}$$
Formal Model

- Episodic MDP with horizon $H$

- In each episode: for $h = 1, ..., H$, learner
  - observes state feature $x_h \in X$ (possibly infinite) (w.l.o.g. $x_1 = x^0$)
  - chooses action $a_h \in A$ (finite & manageable)
  - receives reward $r_h \in \mathbb{R}$ (bounded)

- Learning goal: given $F$ such that $Q^* \in F$, (will relax)
w.p. $1 - \delta$, find policy $\pi$ s.t. $v^* - v^\pi \leq \varepsilon$
using $\text{poly}(|A|, H, \log|F|, 1/\varepsilon, 1/\delta)$ episodes. (can extend to VC-dim)

$$F = \{ f(\cdot ; \theta) : \theta \in \Theta \}$$

[exponential (in $H$)
lower bound exists!]

[Krishnamurthy et al’16]
Finite MDPs
[Kearns & Singh’98]
(small #states)
Zoo of RL Exploration

- Finite MDPs [Kearns & Singh’98] (small #states)
- Metric space [Kakade et al’03] Abstraction [Li’09] (small #abstract states)
- Worst-case construction

\[ \mathcal{F} \]
Zoo of RL Exploration

Finite MDPs [Kearns & Singh’98] (small #states)

Metric space [Kakade et al’03]

Abstraction [Li’09] (small #abstract states)

LQR control [Ibrahimi et al’12] (small #variables)

\[ F = \{ f(\cdot); \}^{\infty} \]

Worst-case construction
Zoo of RL Exploration

Finite MDPs [Kearns & Singh’98] (small #states)

Metric space [Kakade et al’03]
Abstraction [Li’09] (small #abstract states)

LQR control [Ibrahimi et al’12] (small #variables)

\[ F = \{ f(\cdot, \cdot) : 2^{\mathcal{X}} \} \]

Worst-case construction
Zoo of RL Exploration

Finite MDPs [Kearns & Singh’98]
(small #states)

Metric space [Kakade et al’03]
Abstraction [Li’09]
(small #abstract states)

LQR control [Ibrahimi et al’12]
(small #variables)

Hidden state

Worst-case construction

\[ F \subseteq \{ f(\cdot); \in L^2 \} \]
Zoo of RL Exploration

Finite MDPs [Kearns & Singh’98] (small #states)

Metric space [Kakade et al’03]
Abstraction [Li’09] (small #abstract states)

deterministic dynamics +
[Krishnamurthy et al’16]

LQR control [Ibrahimi et al’12] (small #variables)

POMDPs w/ rich observation and reactive value function (small #hidden-states)

Worst-case construction
Zoo of RL Exploration

Finite MDPs [Kearns & Singh’98] (small #states)

LQR control [Ibrahimi et al’12] (small #variables)

Metric space [Kakade et al’03]
Abstraction [Li’09] (small #abstract states)

deterministic dynamics + [Krishnamurthy et al’16]

POMDPs w/ rich observation and reactive value function (small #hidden-states)

MDPs w/ low-rank transition matrix [Barreto et al’11] (small matrix rank)

$P_T|h$

Same setup in PSRs [Littman et al’02] (small system dim.)

Worst-case construction

$\mathcal{F} \{ \ldots \}$
Zoo of RL Exploration

Finite MDPs  
[Kearns & Singh’98]  
(small #states)

Metric space  
[Kakade et al’03]  
Abstraction  
[Li’09]  
(small #abstract states)

LQR control  
[Ibrahimi et al’12]  
(small #variables)

MDPs w/ low-rank transition matrix  
[Barreto et al’11]  
(small matrix rank)

POMDPs w/ rich observation and reactive value function  
[Krishnamurthy et al’16]  
(small #hidden-states)

P_T|h

Same setup in PSRs  
[Littman et al’02]  
(small system dim.)

L

Worst-case construction

[JKALS’16]

• All these settings yield low Bellman rank

P(x’|x,a) = [ ]

Zoo of RL Exploration

- Finite MDPs [Kearns & Singh’98] (small #states)
- Metric space [Kakade et al’03]
- Abstraction [Li’09] (small #abstract states)
- LQR control [Ibrahimi et al’12] (small #variables)
- POMDPs w/ rich observation and reactive value function (small #hidden-states)
- MDPs w/ low-rank transition matrix [Barreto et al’11] (small matrix rank)
- Deterministic dynamics + [Krishnamurthy et al’16]

[JKALS’16]
- All these settings yield low Bellman rank
- Unified algorithm, polynomial guarantee

\[ P_{\mathcal{T}|h} \]
Same setup in PSRs [Littman et al’02] (small system dim.)

Worst-case construction
Defining Bellman rank

Step 1: Average Bellman Error

• Bellman error of $f$ at $(x_h, a_h)$

$$f(x_h, a_h) - \mathbb{E}_{r_h, x_{h+1} | x_h, a_h} \left[ r_h + \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right]$$
Defining Bellman rank
Step 1: Average Bellman Error

• Bellman error of $f$ at $(x_h, a_h)$

$$f(x_h, a_h) - \mathbb{E}_{r_h, x_{h+1} | x_h, a_h} \left[ r_h + \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right]$$

- $Q^*$ has 0 Bellman error for all $(x_h, a_h)$. 
Defining Bellman rank
Step 1: Average Bellman Error

- Bellman error of $f$ at $(x_h, a_h)$
  \[
  f(x_h, a_h) - \mathbb{E}_{r_h, x_{h+1} | x_h, a_h} \left[ r_h + \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right]
  \]
  - $Q^*$ has 0 Bellman error for all $(x_h, a_h)$.

- Average Bellman error of $f$ is the linear combination of its Bellman errors over $(x_h, a_h)$
Defining Bellman rank
Step 1: **Average Bellman Error**

- **Bellman error of** \( f \) **at** \((x_h, a_h)\)

\[
f(x_h, a_h) - \mathbb{E}_{r_h, x_{h+1} | x_h, a_h} \left[ r_h + \max_{a \in A} f(x_{h+1}, a) \right]
\]

- \( Q^* \) **has** 0 Bellman error for all \((x_h, a_h)\).

- **Average Bellman error** of \( f \) **is the linear combination** of its Bellman errors over \((x_h, a_h)\)

- **Weights**: distribution over \( x_h \) induced by policy \( \pi \).

\[
\mathcal{E}^h(f, \pi) := \mathbb{E}_{a_{1:h-1} \sim \pi} \left[ f(x_h, a_h) - r_h - \max_{a \in A} f(x_{h+1}, a) \right]
\]

\[ a_h = \text{arg max } f(x_h, \cdot) \]
Defining Bellman rank

Step 1: Average Bellman Error

• Bellman error of $f$ at $(x_h, a_h)$

$$f(x_h, a_h) - \mathbb{E}_{r_h, x_{h+1}|x_h, a_h} \left[ r_h + \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right]$$

- $Q^*$ has 0 Bellman error for all $(x_h, a_h)$.

• Average Bellman error of $f$ is the linear combination of its Bellman errors over $(x_h, a_h)$

- Weights: distribution over $x_h$ induced by policy $\pi$.

$$\mathcal{E}^h(f, \pi) := \mathbb{E}_{a_1 \sim h-1 \sim \pi} \left[ f(x_h, a_h) - r_h - \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right]$$

$$a_h = \arg \max_{a \in \mathcal{A}} f(x_h, \cdot)$$

- $\mathcal{E}^h(Q^*, \pi) = 0$ for all $\pi$ and $h$.  

Defining Bellman rank
Step 2: Bellman error matrices

$$f \in \mathcal{F}$$

$$\pi \in \Pi_{\mathcal{F}}$$

$$\mathcal{E}^h(f, \pi) :=$$

$$\mathbb{E}_{a_1:h-1 \sim \pi} \left[ f(x_h, a_h) - r_h - \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right]$$
Defining Bellman rank
Step 2: Bellman error matrices

\[ f \in \mathcal{F} \]

\[ \pi \in \Pi_\mathcal{F} \]

class of greedy policies induced from \( F \):
\[ \Pi_\mathcal{F} := \{ x \mapsto \arg \max f(x, \cdot) : f \in \mathcal{F} \} \]

\[ \mathcal{E}^h(f, \pi) := \mathbb{E}_{a_1:h-1 \sim \pi} \left[ f(x_h, a_h) - r_h - \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right] \]
Defining Bellman rank

Step 2: Bellman error matrices

\[ f \in \mathcal{F} \]

\[ \pi \in \Pi_{\mathcal{F}} \]

\[ \mathcal{E}^h(f, \pi) := \mathbb{E}_{a_{1:h-1} \sim \pi} \left[ f(x_h, a_h) - r_h - \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right] \]

Definition: Bellman rank is an uniform upper bound on the rank of matrices \( [\mathcal{E}^h(f, \pi)]_{\pi,f} \) over \( h = 1, 2, \ldots, H \).
Tabular MDP: Bellman rank $\leq \# \text{states}$

$E^h(f, \pi) = \pi \times x$

$E_{a_1:h-1 \sim f, a_h \sim f} [f(x_h, a_h) - r_h - \max_{a \in \mathcal{A}} f(x_{h+1}, a)]$

Bellman error of $f$ on each state

distribution over states induced by $\pi$
“Visual grid-world”: Bellman rank $\leq \# \text{hidden states}$

hidden state

rendered image

value
“Visual grid-world”: Bellman rank $\leq$ # hidden states

$$\mathcal{E}^h(f, \pi) := \mathbb{E}_{a_1 \sim h-1 \sim \pi} \left[ f(x_h, a_h) - r_h - \max_{a \in A} f(x_{h+1}, a) \right]$$
“Visual grid-world”: Bellman rank $\leq \#$ hidden states

$$\mathcal{E}^h(f, \pi) := \mathbb{E}_{a_1: h-1 \sim \pi, a_h \sim f} \left[ f(x_h, a_h) - r_h - \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right]$$
“Visual grid-world”: Bellman rank $\leq \# \text{ hidden states}$

$$
\mathcal{E}^h(f, \pi) := \\
\mathbb{E}_{a_1:h-1 \sim \pi, a_h \sim f}[f(x_h, a_h) - r_h - \max_{a \in \mathcal{A}} f(x_{h+1}, a)]
$$
“Visual grid-world”: Bellman rank $\leq \# \text{ hidden states}$

$$\mathcal{E}^h(f, \pi) := \mathbb{E}_{a_1:h-1 \sim \pi, a_h \sim f}[f(x_h, a_h) - r_h - \max_{a \in A} f(x_{h+1}, a)]$$
“Visual grid-world”: Bellman rank ≤ # hidden states

\[ \mathcal{E}^h(f, \pi) := \mathbb{E}_{a_1, \ldots, a_{h-1} \sim \pi} \left[ f(x_h, a_h) - r_h - \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right] \]
Q*-irrelevant abstractions
Q*-irrelevant abstractions

- Number of abstract states is small
Q*-irrelevant abstractions

• Number of abstract states is small
• Challenge: abstract state does not “block” influence from past
Q*-irrelevant abstractions

- Number of abstract states is small
- Challenge: abstract state does not “block” influence from past
- Witness statistics: for each possible \((x, a, r, x')\)
  \[
  \Pr_{a_1:h-1 \sim \pi}[x_h = x, r_h = r, x_{h+1} = x' \mid \text{do } a_h = a]
  \]
Q*-irrelevant abstractions

- Number of abstract states is small
- Challenge: abstract state does not “block” influence from past
- Witness statistics: for each possible \((x, a, r, x')\)

\[
\Pr_{a_1:h-1 \sim \pi}[x_h = x, r_h = r, x_{h+1} = x' \mid \text{do } a_h = a]
\]
Q*-irrelevant abstractions

- Number of abstract states is small
- Challenge: abstract state does not “block” influence from past
- Witness statistics: for each possible \((x, a, r, x')\)

\[
\Pr_{a_1:h-1 \sim \pi}[x_h = x, r_h = r, x_{h+1} = x' \mid \text{do } a_h = a]
\]

- Dimension: \((\#\text{abstract states})^2 \times (\# \text{actions}) \times (\# \text{possible values for reward})\)
Q*-irrelevant abstractions

- Number of abstract states is small
- Challenge: abstract state does not “block” influence from past
- Witness statistics: for each possible \((x, a, r, x')\)
  \[
  \Pr_{a_1:h-1 \sim \pi} [x_h = x, r_h = r, x_{h+1} = x' \mid \text{do } a_h = a]
  \]
- Dimension: \((\#\text{abstract states})^2 \times (\# \text{actions}) \times (\# \text{possible values for reward})\)
  - Reward can always be discretized (and incur a small error)
Zoo of RL Exploration

Finite MDPs [Kearns & Singh’98] (small #states)

Metric space [Kakade et al’03]
Abstraction [Li’09] (small #abstract states)

LQR control [Ibrahimi et al’12] (small #variables)

MDPs w/ low-rank transition matrix
[Barreto et al’11] (small linear rank)

P(x’|x,a) = x

POMDPs w/ rich observation
and reactive value function
(small #hidden-states)

Same setup in PSRs
[Littman et al’02] (small system dim.)

Worst-case construction

F(X) = {f(·); ✓}2 ⇥
Zoo of RL Exploration

\[ B\text{-}rank \leq \#\text{states} \quad B\text{-}rank \leq \text{poly}(\#\text{abs. states}) \]

\[ P_{T|\theta} \]

\[ B\text{-}rank \leq \text{poly}(\text{system dim.}) \]

\[ B\text{-}rank \leq \#\text{hidden-states} \]

\[ \text{B-rank} \leq \text{transition-matrix rank} \]

\[ \text{F} \}

Worst-case construction

\[ P(x'|x,a) = \]

\[ B\text{-}rank \leq \text{poly}(\#\text{variables}) \]

\[ + \text{deterministic dynamics} \quad \text{[Krishnamurthy et al'16]} \]

\[ \checkmark \]
New algorithm: OLIVE
(Optimism-Led Iterative Value-function Elimination)

\[ F_1 := F. \quad \text{// version space} \quad \text{ (Ignoring statistical slackness parameters)} \]

For iteration \( t = 1, 2, \ldots \)
New algorithm: OLIVE
(Optimism-Led Iterative Value-function Elimination)

\( F_1 := F. \) \hspace{1cm} // \text{version space} \quad \text{(Ignoring statistical slackness parameters)}

For iteration \( t=1, 2, \ldots \)

- Choose \( f_t \) as the \( f \in F_t \) that maximizes

\[
\nu_f := \max_{a \in A} f(x^0, a)
\]
New algorithm: OLIVE
(Optimism-Led Iterative Value-function Elimination)

\[ F_1 := F. \quad \text{// version space} \quad \text{(Ignoring statistical slackness parameters)} \]

For iteration \( t=1, 2, \ldots \)

- Choose \( f_t \) as the \( f \in F_t \) that maximizes \( v_f := \max_{a \in A} f(x^0, a) \)
- **Estimate** the value of \( \pi_t \) — the greedy policy of \( f_t \).
New algorithm: OLIVE
(Optimism-Led Iterative Value-function Elimination)

\( F_1 := F. \) // version space (Ignoring statistical slackness parameters)

For iteration \( t=1, 2, \ldots \)

- Choose \( f_t \) as the \( f \in F_t \) that maximizes \( v_f := \max_{a \in \mathcal{A}} f(x^0, a) \)

- Estimate the value of \( \pi_t \) — the greedy policy of \( f_t \).
  - If \( v^{\pi_t} \geq v_{f_t} \), return \( \pi_t \).
New algorithm: OLIVE
(Optimism-Led Iterative Value-function Elimination)

\[ F_1 := F. \quad \text{// version space} \quad \text{(Ignoring statistical slackness parameters)} \]

For iteration \( t=1, 2, \ldots \)

- Choose \( f_t \) as the \( f \in F_t \) that maximizes \( v_f := \max_{a \in A} f(x^0, a) \)

- Estimate the value of \( \pi_t \) — the greedy policy of \( f_t \).

- If \( v_{\pi_t} \geq v_{f_t} \), return \( \pi_t \).

Estimate by MC evaluation
New algorithm: OLIVE
(Optimism-Led Iterative Value-function Elimination)

\[ F_1 := F. \] // version space (Ignoring statistical slackness parameters)

For iteration \( t=1, 2, \ldots \)

- Choose \( f_t \) as the \( f \in F_t \) that maximizes \( v_f := \max_{a \in A} f(x^0, a) \)
- Estimate the value of \( \pi_t \) — the greedy policy of \( f_t \).
  - If \( v^{\pi_t} \geq v_{f_t} \quad (\geq v_Q^* = v^*) \), return \( \pi_t \).
New algorithm: OLIVE
(Optimism-Led Iterative Value-function Elimination)

\[ F_1 := F. \quad \text{// version space} \quad \text{\small (Ignoring statistical slackness parameters)} \]

For iteration \( t = 1, 2, \ldots \)

- Choose \( f_t \) as the \( f \in F_t \) that maximizes \( v_f := \max_{a \in A} f(x^0, a) \)

- Estimate the value of \( \pi_t \) — the greedy policy of \( f_t \).
  - If \( v^{\pi_t} \geq v_{f_t} \ (\geq v_Q = v^* \), return \( \pi_t \).

- Estimate \( \mathcal{E}^h(f, \pi_t) \) for all \( f, h \).

Bellman error matrix
New algorithm: OLIVE
(Optimism-Led Iterative Value-function Elimination)

\[ F_1 := F. \quad // \text{version space} \quad (\text{Ignoring statistical slackness parameters}) \]

For iteration \( t=1, 2, \ldots \)

- Choose \( f_t \) as the \( f \in F_t \) that maximizes \( v_f := \max_{a \in A} f(x^0, a) \)

- **Estimate** the value of \( \pi_t \) — the greedy policy of \( f_t \).
  - If \( v^{\pi_t} \geq v_{f_t} (\geq v_Q^* = v^*) \), return \( \pi_t \).

- **Estimate** \( \mathcal{E}^h(f, \pi_t) \) for all \( f, h \).

- **Eliminate** \( f \) s.t. \( \mathcal{E}^h(f, \pi_t) \neq 0, \forall h \)

\[ \Rightarrow F_{t+1}. \]
New algorithm: OLIVE  
(Optimism-Led Iterative Value-function Elimination)

\[ F_1 := F. \quad // \text{version space} \quad (\text{Ignoring statistical slackness parameters}) \]

For iteration \( t=1, 2, \ldots \)

- Choose \( f_t \) as the \( f \in F_t \) that maximizes \( v_f := \max_{a \in \mathcal{A}} f(x^0, a) \)
- Estimate the value of \( \pi_t \) — the greedy policy of \( f_t \).
  - If \( v_{\pi_t} \geq v_{f_t} \quad (\geq v_{Q^*} = v^* ) \), return \( \pi_t \).
- Estimate \( \mathcal{E}^h(f, \pi_t) \) for all \( f, h \).
- Eliminate \( f \) s.t. \( \mathcal{E}^h(f, \pi_t) \neq 0, \forall h \)  \[ \Rightarrow F_{t+1}. \]

Bellman error matrix

\( \pi_t \)

\( \neq 0 \quad \neq 0 \)

\[ f \]

\( \neq 0 \quad \neq 0 \)
Sample complexity analysis

For iteration $t=1, 2, ...$

- **Estimate** the value of $\pi_t$ — the greedy policy of $f_t$.

  How many sample trajectories needed?

- **Estimate** $\mathcal{E}^h(f, \pi_t)$ for all $f, h$. 
Sample complexity analysis

For iteration $t=1, 2, \ldots$

Run $\pi_t$ for $O(1/\varepsilon^2)$ episodes — Done.

- **Estimate** the value of $\pi_t$ — the greedy policy of $f_t$.

**How many sample trajectories needed?**

- **Estimate** $\mathcal{E}^h(f, \pi_t)$ for all $f, h$. 

$F \setminus \{f\}$
Sample complexity analysis

For iteration \( t=1, 2, \ldots \)

- **Run** \( \pi_t \) for \( O(1/\varepsilon^2) \) episodes — Done.

- **Estimate** the value of \( \pi_t \) — the greedy policy of \( f_t \).

- **Estimate** \( \mathcal{E}^h(f, \pi_t) \) for all \( f, h \).

\[
\mathbb{E}_{a_1:h-1 \sim \pi_t, a_h \sim \mathbb{F}[f \cdots]} \]

How many sample trajectories needed?
Sample complexity analysis

For iteration $t=1, 2, \ldots$

- **Run** $\pi_t$ for $O(1/\varepsilon^2)$ episodes — Done.

- Estimate the value of $\pi_t$ — the greedy policy of $f_t$.

- Estimate $\mathcal{E}^h(f, \pi_t)$ for all $f, h$.  
  $$\mathbb{E}_{a_{1:h-1} \sim \pi_t, a_h \sim f} [f \cdot \cdot \cdot]$$

- Naive: collect data with $a_{1:h-1} \sim \pi_t, a_h \sim f$ for each $f$
Sample complexity analysis

For iteration $t=1, 2, \ldots$

- **Run** $\pi_t$ for $O(1/\varepsilon^2)$ episodes — Done.

- **Estimate** the value of $\pi_t$ — the greedy policy of $f_t$.

- **Estimate** $\mathcal{E}^h(f, \pi_t)$ for all $f, h$.

  \[ \mathbb{E}_{a_1:h-1 \sim \pi_t, a_h \sim f} \left[ f \cdots \right] \]

  - Naive: collect data with $a_{1:h-1} \sim \pi_t, a_h \sim f$ for each $f$
  - $|F|$ samples — too many
Sample complexity analysis

For iteration $t = 1, 2, \ldots$

- Run $\pi_t$ for $O(1/\varepsilon^2)$ episodes — Done.

- **Estimate** the value of $\pi_t$ — the greedy policy of $f_t$.

- **Estimate** $\mathcal{E}^h(f, \pi_t)$ for all $f, h$.

How many sample trajectories needed?

- Naive: collect data with $a_{1:h-1} \sim \pi_t$, $a_h \sim f$ for each $f$
- $|F|$ samples — too many
- Instead: $a_{1:h-1} \sim \pi_t$, $a_h \sim \text{Unif}(A)$ & Importance Sampling
Sample complexity analysis

For iteration $t=1, 2, \ldots$

- Run $\pi_t$ for $O(1/\varepsilon^2)$ episodes — Done.

- **Estimate** the value of $\pi_t$ — the greedy policy of $f_t$.

  How many sample trajectories needed?

- **Estimate** $\mathcal{E}^h(f, \pi_t)$ for all $f, h$.  

  $\mathbb{E}_{a_{1:h-1} \sim \pi_t, a_h \sim f} [f \cdots]$ 

  - Naive: collect data with $a_{1:h-1} \sim \pi_t, a_h \sim f$ for each $f$ 
  - $|F|$ samples — too many 
  - Instead: $a_{1:h-1} \sim \pi_t, a_h \sim \text{Unif}(A)$ & Importance Sampling 
  - 1 sample of size $O(|A| \log |F|/\varepsilon^2)$ — works for all $f$ simultaneously
Sample complexity analysis

For iteration $t=1, 2, \ldots$  

How many iterations???

Run $\pi_t$ for $O(1/\varepsilon^2)$ episodes — Done.

- Estimate the value of $\pi_t$ — the greedy policy of $f_t$.

How many sample trajectories needed?

- Estimate $\mathcal{E}^h(f, \pi_t)$ for all $f, h$.

\[ \mathbb{E}_{a_{1:h-1} \sim \pi_t, a_h \sim f} [f \cdot \cdot \cdot] \]

- Naive: collect data with $a_{1:h-1} \sim \pi_t, a_h \sim f$ for each $f$
- $|F|$ samples — too many
- Instead: $a_{1:h-1} \sim \pi_t, a_h \sim \text{Unif}(A)$ & Importance Sampling
- 1 sample of size $O(|A| \log |F|/\varepsilon^2)$ — works for all $f$ simultaneously
Sample complexity analysis

Claim: If no statistical errors, $\# \text{iterations} \leq \text{Bellman rank}$. 
Sample complexity analysis

Claim: If no statistical errors, \#iterations \leq \text{Bellman rank}.
Sample complexity analysis

Claim: If no statistical errors, \( \# \text{iterations} \leq \text{Bellman rank}. \)

• All surviving \( f \) have all-0 columns so far
Sample complexity analysis

Claim: If no statistical errors, \( \# \text{iterations} \leq \text{Bellman rank} \).

- All surviving \( f \) have all-0 columns so far
- Will show: some \( f \) has “\( \neq 0 \)” in the next iteration

Bellman error matrix
Sample complexity analysis

Claim: If no statistical errors, \# \textit{iterations} \leq \textit{Bellman rank}.

- All surviving \( f \) have all-0 columns so far
- Will show: some \( f \) has “\( \neq 0 \)” in the next iteration

Bellman error matrix
Sample complexity analysis

Claim: If no statistical errors, \# iterations ≤ Bellman rank.

• All surviving \( f \) have all-0 columns so far
• Will show: some \( f \) has “\( \neq 0 \)” in the next iteration
• Then: linearly independent rows ⇒ \# iterations ≤ matrix rank

\[ f \]

\[
\begin{bmatrix}
\pi_t \\
\end{bmatrix}
\begin{bmatrix}
\neq 0 & \neq 0 \\
\neq 0 & \neq 0 \\
\end{bmatrix}
\]
Bellman error matrix
Claim: If no statistical errors, \( \# \text{iterations} \leq \text{Bellman rank} \).

- All surviving \( f \) have all-0 columns so far
- Will show: some \( f \) has “\( \neq 0 \)” in the next iteration
- Then: linearly independent rows \( \Rightarrow \# \text{iterations} \leq \text{matrix rank} \)

\( f_t \) has “\( \neq 0 \)” unless terminate:
(recall \( \pi_t \) is greedy wrt \( f_t \))

\[
v_{f_t} - v^{\pi_t} = \sum_{h=1}^{H} \mathcal{E}^h(f_t, \pi_t)
\]
Sample complexity analysis

Claim: If no statistical errors, \( \# \text{iterations} \leq \text{Bellman rank}. \)

- All surviving \( f \) have all-0 columns so far
- Will show: some \( f \) has “\( \neq 0 \)” in the next iteration
- Then: linearly independent rows \( \Rightarrow \# \text{iterations} \leq \text{matrix rank} \)

\( f_t \) has “\( \neq 0 \)” unless terminate:
(recall \( \pi_t \) is greedy wrt \( f_t \))

\[
0 < v_{f_t} - v^{\pi_t} = \sum_{h=1}^{H} E^h(f_t, \pi_t)
\]

Optimized: \( v_{f_t} \geq v_{Q^*} = v^* \)

Bellman error matrix
Sample complexity of OLIVE

\[\pi_t \leq \varphi (\geq -\varphi)\]

\[f\text{ survives if } E(f, \varphi, h) > 0, \quad 8h\]
Sample complexity of OLIVE

$f$ survives if $E(f, \pi_t, h) > 0$, $M = 2$
Sample complexity of OLIVE

\[ \leq \phi (\geq -\phi) \]

\[ f \text{ survives if } x \leq \phi (\geq -\phi) \]
Sample complexity of OLIVE

**Theorem:** If $Q^* \in \mathcal{F}$, w.p. $\geq 1-\delta$, OLIVE returns a $\varepsilon$-optimal policy after acquiring the following number of trajectories

$$\tilde{O}\left(\frac{M^2 H^3 |A|}{\epsilon^2} \log(|\mathcal{F}|/\delta)\right)$$

![Diagram](image)
Other Related Work

• **Sample complexity of AVI-type methods**
  (e.g., Munos 2003; Antos et al., 2008; Munos & Szepesevari 2008)
  
  - batch setting, assume exploratory dataset
  - we focus on exploration
Other Related Work

• **Sample complexity of AVI-type methods** (e.g., Munos 2003; Antos et al., 2008; Munos & Szepesvari 2008)
  - batch setting, assume exploratory dataset
  - we focus on exploration

• **PEGASUS** (Ng & Jordan, 2000)
  - amount of randomness used is polynomial in statistical complexity of $F$
  - requires full control over pseudo-randomness for state transition
Other Related Work

• Sample complexity of AVI-type methods
  (e.g., Munos 2003; Antos et al., 2008; Munos & Szepesvari 2008)
  - batch setting, assume exploratory dataset
  - we focus on exploration

• PEGASUS (Ng & Jordan, 2000)
  - amount of randomness used is polynomial in statistical complexity of $F$
  - requires full control over pseudo-randomness for state transition

• Eluder dimension & OCP (Wen & Van Roy, 2013)
  - requires fully deterministic dynamics
  - eluder dimension shown to be small for linear / quadratic functions
Bellman Equations revisited

$$\mathbb{E}_{a_{1:h-1} \sim \pi', \ a_h \sim f} [f(x_h, a_h) - r_h - \max_{a \in A} f(x_{h+1}, a)] = 0$$
Bellman Equations revisited

$$
\mathbb{E}_{a_1:h-1 \sim \pi', \ a_h \sim f} \left[ f(x_h, a_h) - r_h - \max_{a \in \mathcal{A}} f(x_{h+1}, a) \right] = 0
$$

• $f$ on non-greedy actions never used!
Bellman Equations revisited

\[ \mathbb{E}_{a_1: h-1 \sim \pi'} [g(x_h) - r_h - g(x_{h+1})] = 0 \]

- \( f \) on non-greedy actions never used!
- Reparametrize: \( f \Rightarrow (g, \pi); F \Rightarrow G, \Pi. \)
Bellman Equations revisited

\[ \mathbb{E}_{a_{1:h-1} \sim \pi', a_h \sim \pi} [g(x_h) - r_h - g(x_{h+1})] = 0 \]

- \( f \) on non-greedy actions never used!
- Reparametrize: \( f \Rightarrow (g, \pi); F \Rightarrow G, \Pi \).
- Bellman equations for policy evaluation
Bellman Equations revisited

\[ E_{a_1: h-1 \sim \pi'} [g(x_h) - r_h - g(x_{h+1})] = 0 \]

- \( f \) on non-greedy actions never used!
- Reparametrize: \( f \rightarrow (g, \pi); F \rightarrow G, \Pi \).
- Bellman equations for policy evaluation
  - Even if \( \pi^* \not\in \Pi \), can still compete with any \( \pi \in \Pi \) whose policy-specific value function is (approx.) in \( G \)
Bellman Equations revisited

\[ E_{a_1:h-1 \sim \pi'} \left[ g(x_h) - r_h - g(x_{h+1}) \right] = 0 \]

- \( f \) on non-greedy actions never used!
- Reparametrize: \( f \Rightarrow (g, \pi); \; F \Rightarrow G, \Pi \).
- Bellman equations for policy evaluation
  - Even if \( \pi^* \notin \Pi \), can still compete with any \( \pi \in \Pi \)
    whose policy-specific value function is (approx.) in \( G \)
  - Allow infinite classes with VC-type dimensions
Bellman Equations revisited

\[ \mathbb{E}_{a_1:h-1 \sim \pi', a_h \sim \pi}[g(x_h) - r_h - g(x_{h+1})] = 0 \]

- What happens if \( x_h \) is not sufficient statistics of history?
  - \( \times \) Standard Bellman equation (state-wise) no longer makes sense
Bellman Equations revisited

$$\mathbb{E}_{a_1:h-1 \sim \pi', a_h \sim \pi} [g(x_h) - r_h - g(x_{h+1})] = 0$$

- What happens if $x_h$ is not sufficient statistics of history?
  - $\times$ Standard Bellman equation (state-wise) no longer makes sense
- Our Bellman equation (distribution-wise) is still well-defined!
  - $\checkmark$ Value-based RL framework without sufficient statistics
Bellman Equations revisited

\[ \mathbb{E}_{a_1:h-1 \sim \pi', a_h \sim \pi} \left[ g(x_h) - r_h - g(x_{h+1}) \right] = 0 \]

- What happens if \( x_h \) is not sufficient statistics of history?
  - \( \textbf{X} \) Standard Bellman equation (state-wise) no longer makes sense
- Our Bellman equation (distribution-wise) is still well-defined!
  - \( \checkmark \) Value-based RL framework without sufficient statistics
- New framework: Contextual Decision Processes (CDPs)
  - \( \checkmark \) Everything is agnostic, learning by competition!
Detailed Analysis (with Statistical Errors)
\[ h_1 \xrightarrow{\varepsilon} h_2 \]

\[
\rho(x h) r h g(x h + 1) = 0
\]
\[ g \]

\[ \pi_{t-1} = \pi_{t-1} \]

\[ M \]

\[ g \]

\[ M=2 \]
$M = 2$

$\pi_{t-1} \xleftarrow{\cdot} \pi_{t-1}$

$g(\pi_{t-1}) = 0$

$X$
\[ \phi \text{ controlled by sample size} \]
key observation: 

\[ \langle \vec{r}, \vec{g} \rangle \] and \[ \langle \vec{r}, \vec{h} \rangle \] are roughly orthogonal

\[ \phi \text{ controlled by sample size} \]
inefficient exploration

- new distribution is similar to previous ones
- area of while space shrinks slowly
inefficient exploration

- new distribution is similar to previous ones
- area of while space shrinks slowly
inefficient exploration

- new distribution is similar to previous ones
- area of white space shrinks slowly

efficient exploration

- new distribution is different from previous ones
- area of white space shrinks quickly
efficient exploration

**Algorithm**
- new distribution is different from previous ones

**Analysis**
- area of while space shrinks quickly
Pick \((g, \pi)\) that (1) obey the Bellman equation constraints so far, (2) \(g\) is optimistic. Then explore with \(\pi\).

**key observation:**
- and \(\pi_{t-1}\) are roughly orthogonal
Pick \((g, \pi)\) that (1) obey the Bellman equation constraints so far, (2) \(g\) is optimistic. Then explore with \(\pi\).

Key observation:
- \(\rightarrow\) and \(\longrightarrow\) are roughly orthogonal
- \(\langle \rightarrow, \longrightarrow \rangle\) is large (parallel)
- \(\longrightarrow\) and \(\longrightarrow\) are orthogonal
Pick \((g, \pi)\) that (1) obey the Bellman equation constraints so far, (2) \(g\) is optimistic. Then explore with \(\pi\).

Lemma: for any \((g, \pi)\),

\[
g(x^0) - v^\pi = \sum_{h=1}^{H} \mathbb{E}_{a_1:h-1 \sim \pi, a_h \sim \pi} \left[ g(x_h) - r_h - g(x_{h+1}) \right]
\]

key observation:
- and \(--\rightarrow\) are roughly orthogonal
- \(\langle \rightarrow, \phantom{--\rightarrow} \rangle\) is large (parallel)
- \(--\rightarrow\) and \(--\rightarrow\) are orthogonal
Pick \((g, \pi)\) that (1) obey the Bellman equation constraints so far, (2) \(g\) is optimistic. Then explore with \(\pi\).

**Lemma:** for any \((g, \pi)\),

\[
g(x^0) - v^\pi = \sum_{h=1}^{H} \langle \rightarrow , \cdots \rightarrow \rangle \text{ at level } h
\]

Key observation:
- \(\rightarrow\) and \(\cdots \rightarrow\) are roughly orthogonal
- \(\langle \rightarrow , \cdots \rightarrow \rangle\) is large (parallel)
- \(\cdots \rightarrow\) and \(\cdots \rightarrow\) are orthogonal
$M=2$
$M=2$
Adaptation of [Todd, 1982]: Ellipsoid volume shrinks exponentially if

\[ |\langle \rightarrow, \ldots \rangle| \geq 3\sqrt{M} \times 2\phi \]
Adaptation of [Todd, 1982]:
Ellipsoid volume shrinks exponentially if

$$|\langle \rightarrow, \rightarrow \rangle| \geq 3\sqrt{M} \times 2\phi$$

controlled by sub-optimality
controlled by sample size
OLIVE requires solving a constrained optimization problem

- $f_t \in \mathcal{F}_t \iff f \in \mathcal{F}, \mathcal{E}^h(f, \pi_{t'}) \neq 0, \forall h \in [H], t' \in [t - 1]$
- $f_t = \max v_f$, subject to the constraints.
Computational Efficiency
[Dann+JKALS, arXiv’18]

• OLIVE requires solving a constrained optimization problem
  • $f \in F_t \iff f \in F, \mathcal{E}^h(f, \pi_{t'}) \neq 0, \forall h \in [H], t' \in [t - 1]$
  • $f_t = \max v_f$, subject to the constraints.

• How to access $F$ (or $G, \Pi$)?
Computational Efficiency
[Dann+JKALS, arXiv’18]

• OLIVE requires solving a constrained optimization problem
  • \( f \in \mathcal{F}_t \iff f \in \mathcal{F}, \mathcal{E}^h(f, \pi_{t'}) \neq 0, \forall h \in [H], t' \in [t-1] \)
  • \( f_t = \max v_f \), subject to the constraints.

• How to access \( F \) (or \( G, \Pi \))?  
  • Oracles. E.g.,
Computational Efficiency
[Dann+JKALS, arXiv’18]

- OLIVE requires solving a constrained optimization problem
  - $f \in \mathcal{F}_t \iff f \in \mathcal{F}, \mathcal{E}^h(f, \pi_{t'}) \neq 0, \forall h \in [H], t' \in [t - 1]$  
  - $f_t = \max v_f$, subject to the constraints.

- How to access $F$ (or $G, \Pi$)?
  - Oracles. E.g.,
    - Cost-sensitive Classification for $\Pi \subset (X \rightarrow A)$
      
      Given $\{(x^i \in X, c^i \in R^A)\}_{i \in [n]}$, oracle minimizes $\sum_{i=1}^{n} c^i(\pi(x^i))$
Computational Efficiency
[Dann+JKALS, arXiv’18]

• OLIVE requires solving a constrained optimization problem
  • $f \in \mathcal{F}_t \iff f \in \mathcal{F}, \mathcal{E}^h(f, \pi_{t'}) \neq 0, \forall h \in [H], t' \in [t - 1]$
  • $f_t = \max v_f$, subject to the constraints.

• How to access $F$ (or $G, \Pi$)?
  • Oracles. E.g.,
    • Cost-sensitive Classification for $\Pi \subset (X \rightarrow A)$
      Given $\{(x^i \in X, c^i \in R^A)\}_{i \in [n]}$, oracle minimizes $\sum_{i=1}^n c^i(\pi(x^i))$
    • Linear optimization, squared-loss regression for $G \subset (X \rightarrow R)$
Computational Efficiency
[Dann+JKALS, arXiv’18]

• OLIVE requires solving a constrained optimization problem
  • $f \in \mathcal{F}_t \iff f \in \mathcal{F}, \mathcal{E}^h(f, \pi_{t'}) \neq 0, \forall h \in [H], t' \in [t - 1]$
  • $f_t = \max v_f$, subject to the constraints.

• How to access $F$ (or $G, \Pi$)?
  • Oracles. E.g.,
    • Cost-sensitive Classification for $\Pi \subset (X \to A)$
      Given $\{(x^i \in X, c^i \in R^A)\}_{i \in [n]}$, oracle minimizes $\sum_{i=1}^{n} c^i(\pi(x^i))$
    • Linear optimization, squared-loss regression for $G \subset (X \to R)$

• Can we reduce the computation of OLIVE to oracles?
Computational Efficiency
[Dann+JKALS, arXiv’18]

- No polynomial reduction exists
Computational Efficiency
[Dann+JKALS, arXiv’18]

• No polynomial reduction exists
  • NP-hard even in tabular MDPs
Computational Efficiency
[Dann+JKALS, arXiv’18]

• No polynomial reduction exists
  • NP-hard even in tabular MDPs
  • ERM also NP-hard — “absorbs” hardness?
Computational Efficiency
[Dann+JKALS, arXiv'18]

- No polynomial reduction exists
  - **NP-hard** even in tabular MDPs
  - ERM also NP-hard — “absorbs” hardness?
  - Common oracles are **efficient** in the tabular case
    i.e., $|X|$ has finite cardinality, $\Pi = X \rightarrow A$
Computational Efficiency
[Dann+JKALS, arXiv’18]

• No polynomial reduction exists
  • **NP-hard** even in tabular MDPs
  • ERM also NP-hard — “absorbs” hardness?
  • Common oracles are **efficient** in the tabular case
    i.e., $|X|$ has finite cardinality, $\Pi = X \rightarrow A$

• Not game-over
  • Algorithm specific result
Computational Efficiency
[Dann+JKALS, arXiv’18]

• No polynomial reduction exists
  • NP-hard even in tabular MDPs
  • ERM also NP-hard — “absorbs” hardness?
• Common oracles are efficient in the tabular case
  i.e., \(|X|\) has finite cardinality, \(\Pi = X \rightarrow A\)

• Not game-over
  • Algorithm specific result
  • In more restricted setting, new algorithm efficient both statistically and computationally

Deterministic dynamics + [Krishnamurthy et al’16]
POMDPs w/ rich observation and reactive value function (small #hidden-states)
Summary

- New complexity measure, **Bellman rank**, that unifies many RL settings where exploration is tractable
tabular, reactive POMDPs / PSRs, low-rank MDPs, LQRs…
Summary

• New complexity measure, **Bellman rank**, that unifies many RL settings where exploration is tractable
  tabular, reactive POMDPs / PSRs, low-rank MDPs, LQRs…

• New algorithm **OLIVE**: polynomial sample complexity
Summary

• New complexity measure, **Bellman rank**, that unifies many RL settings where exploration is tractable
  tabular, reactive POMDPs / PSRs, low-rank MDPs, LQRs…

• New algorithm **OLIVE**: polynomial sample complexity

• New conceptual framework: **Contextual Decision Processes**
Summary

• New complexity measure, Bellman rank, that unifies many RL settings where exploration is tractable
  tabular, reactive POMDPs / PSRs, low-rank MDPs, LQRs…

• New algorithm OLIVE: polynomial sample complexity

• New conceptual framework: Contextual Decision Processes

• Next step: computational efficiency
  • Solved for reactive POMDPs with deterministic hidden dynamics
  • Some negative results for the stochastic case; still open!
Summary

- New complexity measure, **Bellman rank**, that unifies many RL settings where exploration is tractable
  tabular, reactive POMDPs / PSRs, low-rank MDPs, LQRs…
- New algorithm **OLIVE**: polynomial sample complexity
- New conceptual framework: **Contextual Decision Processes**
- Next step: computational efficiency
  - Solved for reactive POMDPs with deterministic hidden dynamics
  - Some negative results for the stochastic case; still open!


[JKALS] ICML-17. **CDPs with low Bellman rank are PAC-Learnable.**