Model-free Approaches to Robust Markov Decision Processes

Wenhao Yang

Stanford Univeristy

2023.10.17

What is Robust RL?

- Two environments: A and B.
- Optimal policy in A may be sub-optimal/bad in B.
- Can we design a conservative policy?

What is Robust MDPs?

- Instance parameters: $\langle \mathcal{S}, \mathcal{A}, R, P^*, \gamma \rangle$.

What is Robust MDPs?

- Instance parameters: $\langle \mathcal{S}, \mathcal{A}, R, P^*, \gamma \rangle$.
- Uncertainty set: $\mathcal{P} = \bigotimes_{s,a} \mathcal{P}_{s,a}$,

What is Robust MDPs?

- Instance parameters: $\langle \mathcal{S}, \mathcal{A}, R, P^*, \gamma \rangle$.
- Uncertainty set: $\mathcal{P} = \bigotimes_{s,a} \mathcal{P}_{s,a}$,

$$\mathcal{P}_{s,a} := \left\{ P(\cdot|s,a) \in \Delta(\mathcal{S}) \middle| \sum_{s' \in \mathcal{S}} f\left(\frac{P(s'|s,a)}{P^*(s'|s,a)} \right) P^*(s'|s,a) \leq \rho \right\}.$$

What is Robust MDPs?

- Instance parameters: $\langle \mathcal{S}, \mathcal{A}, R, P^*, \gamma \rangle$.
- Uncertainty set: $\mathcal{P} = \bigotimes_{s,a} \mathcal{P}_{s,a}$,

$$\mathcal{P}_{s,a} := \left\{ P(\cdot|s,a) \in \Delta(\mathcal{S}) \middle| \sum_{s' \in \mathcal{S}} f\left(\frac{P(s'|s,a)}{P^*(s'|s,a)} \right) P^*(s'|s,a) \le \rho \right\}.$$

- Robust value function with given policy π:

What is Robust MDPs?

- Instance parameters: $\langle \mathcal{S}, \mathcal{A}, R, P^*, \gamma \rangle$.
- Uncertainty set: $\mathcal{P} = \bigotimes_{s,a} \mathcal{P}_{s,a}$,

$$\mathcal{P}_{s,a} := \left\{ P(\cdot|s,a) \in \Delta(\mathcal{S}) \,\middle|\, \sum_{s' \in \mathcal{S}} f\left(\frac{P(s'|s,a)}{P^*(s'|s,a)} \right) P^*(s'|s,a) \leq \rho \right\}.$$

- Robust value function with given policy π:

$$V^{\pi}_{\text{rob,c}}(s) := \inf_{P \in \mathcal{P}} V^{\pi}_{P}(s).$$

What is Robust MDPs?

- Instance parameters: $\langle \mathcal{S}, \mathcal{A}, R, P^*, \gamma \rangle$.
- Uncertainty set: $\mathcal{P} = \bigotimes_{s,a} \mathcal{P}_{s,a}$,

$$\mathcal{P}_{s,a} := \left\{ P(\cdot|s,a) \in \Delta(\mathcal{S}) \middle| \sum_{s' \in \mathcal{S}} f\left(\frac{P(s'|s,a)}{P^*(s'|s,a)} \right) P^*(s'|s,a) \leq \rho \right\}.$$

- Robust value function with given policy π:

$$V^{\pi}_{\text{rob,c}}(s) := \inf_{P \in \mathcal{P}} V^{\pi}_{P}(s).$$

- Optimal robust value function $V^{*}_{\text{rob,c}}(s) := \max_{\pi} V^{\pi}_{\text{rob,c}}(s)$.

Why Robust MDPs?

- Why care robust MDPs?

Why Robust MDPs?

- Why care robust MDPs?
- A large ρ can reduce sample complexity. [YZZ22]

Why Robust MDPs?

- Why care robust MDPs?
- A large ρ can reduce sample complexity. [YZZ22]

Theorem [YZZ22]

There exists a class of MDPs, given that $f(t) = (t-1)^2$, for every (ε, δ)-correct RL algorithm $\mathscr{A}$, the total number of samples needs to be at least:

$$\widetilde{\Omega}\left(\frac{|\mathcal{S}||\mathcal{A}|}{\varepsilon^2(1-\gamma)^2}\min\left\{\frac{1}{1-\gamma},\frac{1}{\rho}\right\}\right). \tag{1}$$

How to solve Robust MDPs?

- $V^*_{\text{rob,c}}$ still satisfies Bellman equation:

How to solve Robust MDPs?

- $V^*_{\text{rob,c}}$ still satisfies Bellman equation:

$$V^*_{\text{rob,c}}(s) = \max_a \left(R(s,a) + \gamma \inf_{P_{s,a}\in\mathcal{P}_{s,a}} \sum_{s'} P(s'|s,a) V^*_{\text{rob,c}}(s') \right)$$
$$:= \mathcal{T}_{\text{rob,c}} V^*_{\text{rob,c}}(s).$$

How to solve Robust MDPs?

- $V^*_{\text{rob,c}}$ still satisfies Bellman equation:

$$V^*_{\text{rob,c}}(s) = \max_a \left(R(s,a) + \gamma \inf_{P_{s,a} \in \mathcal{P}_{s,a}} \sum_{s'} P(s'|s,a) V^*_{\text{rob,c}}(s') \right)$$
$$:= \mathcal{T}_{\text{rob,c}} V^*_{\text{rob,c}}(s).$$

- If we know:
  - A good estimiation of P^*,
  - Solution of inner optimization problem $\inf_{P_{s,a} \in \mathcal{P}_{s,a}} \sum_{s'} P(s'|s,a) V(s')$ for any given V,

How to solve Robust MDPs?

- $V^*_{\text{rob,c}}$ still satisfies Bellman equation:

$$V^*_{\text{rob,c}}(s) = \max_a \left(R(s,a) + \gamma \inf_{P_{s,a} \in \mathcal{P}_{s,a}} \sum_{s'} P(s'|s,a) V^*_{\text{rob,c}}(s') \right)$$
$$:= \mathcal{T}_{\text{rob,c}} V^*_{\text{rob,c}}(s).$$

- If we know:
  - A good estimiation of P^*,
  - Solution of inner optimization problem $\inf_{P_{s,a} \in \mathcal{P}_{s,a}} \sum_{s'} P(s'|s,a)V(s')$ for any given V,

near-optimal robust value function can be obtained with efficient sample complexity.[YZZ22]

Problems on Current Setting.

- Memory space $\mathcal{O}(|\mathcal{S}|^2|\mathcal{A}|)$. (Storing $\widehat{P}$)

Problems on Current Setting.

- Memory space $\mathcal{O}(|\mathcal{S}|^2|\mathcal{A}|)$. (Storing $\widehat{P}$)
- Computation complexity of inner optimization problem enlarges with instance size.

Problems on Current Setting.

- Memory space $\mathcal{O}(|\mathcal{S}|^2|\mathcal{A}|)$. (Storing $\widehat{P}$)
- Computation complexity of inner optimization problem enlarges with instance size.
- Question: can we design a model-free algorithm with efficient sample complexity (including the complexity of inner optimization problem)?

1. Introduction
2. Model-free robust MDPs
3. Conclusion
4. Reference

Q-learning

- Q-learning updating rule:

Q-learning

- Q-learning updating rule:

$$Q_{t+1}(s,a) = (1-\alpha_t)Q_t(s,a) + \alpha_t \widehat{\mathcal{T}} Q_t(s,a), \tag{2}$$

Q-learning

- Q-learning updating rule:

$$Q_{t+1}(s,a) = (1-\alpha_t)Q_t(s,a) + \alpha_t \widehat{\mathcal{T}} Q_t(s,a), \tag{2}$$

where $\widehat{\mathcal{T}} Q_t(s,a) = r_t(s,a) + \gamma \max_a Q_t(s'_t, a)$

Q-learning

- Q-learning updating rule:

$$Q_{t+1}(s,a) = (1-\alpha_t)Q_t(s,a) + \alpha_t \widehat{\mathcal{T}} Q_t(s,a), \tag{2}$$

where $\widehat{\mathcal{T}} Q_t(s,a) = r_t(s,a) + \gamma \max_a Q_t(s'_t, a)$ and satisfies $\mathbb{E}[\widehat{\mathcal{T}} Q_t | \mathcal{F}_t] = \mathcal{T} Q_t = R(s,a) + \gamma \mathbb{E}_P \max_{a'} Q_t(s', a)$.

Q-learning

- Q-learning updating rule:

$$Q_{t+1}(s,a) = (1-\alpha_t)Q_t(s,a) + \alpha_t \widehat{\mathcal{T}} Q_t(s,a), \quad (2)$$

where $\widehat{\mathcal{T}} Q_t(s,a) = r_t(s,a) + \gamma \max_a Q_t(s_t', a)$ and satisfies $\mathbb{E}[\widehat{\mathcal{T}} Q_t | \mathcal{F}_t] = \mathcal{T} Q_t = R(s,a) + \gamma \mathbb{E}_P \max_{a'} Q_t(s', a)$.

- Subtracting Q^* each side:

Q-learning

- Q-learning updating rule:

$$Q_{t+1}(s,a) = (1-\alpha_t)Q_t(s,a) + \alpha_t \widehat{\mathcal{T}} Q_t(s,a), \tag{2}$$

where $\widehat{\mathcal{T}} Q_t(s,a) = r_t(s,a) + \gamma \max_a Q_t(s_t', a)$ and satisfies $\mathbb{E}[\widehat{\mathcal{T}} Q_t | \mathcal{F}_t] = \mathcal{T} Q_t = R(s,a) + \gamma \mathbb{E}_P \max_{a'} Q_t(s', a)$.

- Subtracting Q^* each side:

$$\begin{aligned} Q_{t+1} - Q^* =& (1-\alpha_t)(Q_t - Q^*) \\ &+ \alpha_t(\widehat{\mathcal{T}} Q_t - \mathcal{T} Q_t) + \alpha_t(\mathcal{T} Q_t - \mathcal{T} Q^*) \end{aligned} \tag{3}$$

Theorem [Wai19, LYZJ21]

For a sequence $X_{t+1} = (1-\alpha_t)X_t + \alpha_t Y_t + \alpha_t \delta_t$, where $\{Y_t\}_{t\geq 0}$ is a martingale difference, $(1-\alpha_t)\alpha_{t-1} \leq \alpha_t$, and $\sum_{t=0}^{T-1} \delta_t = o\left(1/\alpha_T\right)$, then X_T satisfies:

$$\mathbb{E}|X_T| \leq \widetilde{\mathcal{O}}\left(\sqrt{\alpha_T} + \alpha_T \sum_{t=0}^{T-1} \delta_t\right) \tag{4}$$

Theorem [Wai19, LYZJ21]

For a sequence $X_{t+1} = (1-\alpha_t)X_t + \alpha_t Y_t + \alpha_t \delta_t$, where $\{Y_t\}_{t\geq 0}$ is a martingale difference, $(1-\alpha_t)\alpha_{t-1} \leq \alpha_t$, and $\sum_{t=0}^{T-1} \delta_t = o\left(1/\alpha_T\right)$, then X_T satisfies:

$$\mathbb{E}|X_T| \leq \widetilde{\mathcal{O}}\left(\sqrt{\alpha_T} + \alpha_T \sum_{t=0}^{T-1} \delta_t\right) \tag{4}$$

- Convergence rate is guarateed [Wai19]:

$$\mathbb{E}\left\|Q_T - Q^*\right\|_\infty = \widetilde{\mathcal{O}}\left(\sqrt{\alpha_T}\right) \tag{5}$$

- For robust MDPs, noting the robust Bellman operator:

$$\mathcal{T}_{\mathsf{rob,c}} Q = R(s,a) + \gamma \inf_{D_f(P \| P^*_{s,a}) \le \rho} \mathbb{E}_{s' \sim P} \max_a Q(s', a) \tag{6}$$

- For robust MDPs, noting the robust Bellman operator:

$$\mathcal{T}_{\text{rob,c}} Q = R(s,a) + \gamma \inf_{D_f(P \| P^*_{s,a}) \le \rho} \mathbb{E}_{s' \sim P} \max_a Q(s', a) \tag{6}$$

- Non-linear functional of expectation.

Key observation

- Simplify the notation of inner optimization problem:

Key observation

- Simplify the notation of inner optimization problem:

$$\inf_{D_f(P\|P^*)\leq\rho} \sum_i P_i V_i.$$

Key observation

- Simplify the notation of inner optimization problem:

$$\inf_{D_f(P\|P^*)\leq\rho} \sum_i P_i V_i.$$

- Dual form:

$$\sup_{\lambda\geq 0, \eta\in\mathbb{R}} -\lambda \sum_i P_i^* f^*\left(\frac{\eta - V_i}{\lambda}\right) - \lambda\rho + \eta,$$

where $f^*(t) = \sup_{s\geq 0}(st - f(s))$.

Key observation

- Simplify the notation of inner optimization problem:

$$\inf_{D_f(P\|P^*)\le\rho}\sum_i P_iV_i.$$

- Dual form:

$$\sup_{\lambda\ge0,\eta\in\mathbb{R}} -\lambda\sum_i P_i^* f^*\left(\frac{\eta-V_i}{\lambda}\right)-\lambda\rho+\eta,$$

where $f^*(t)=\sup_{s\ge0}(st-f(s))$.

- How to construct a good estimator for $\mathcal{T}_r V$ with a given V and $\mathcal{O}(1)$ samples? (specifically, unbiased)

Key observation

- Problem is equivalent to construct a good estimator for:

Key observation

- Problem is equivalent to construct a good estimator for:

$$\sup_{\theta} \mathbb{E}_P f(X;\theta). \tag{7}$$

Key observation

- Problem is equivalent to construct a good estimator for:

$$\sup_{\theta} \mathbb{E}_P f(X;\theta). \tag{7}$$

- But $\mathbb{E}\sup_{\theta} \frac{1}{n}\sum_{i=1}^{m} f(X_i;\theta) \neq \sup_{\theta} \mathbb{E}_P f(X;\theta)$.

Key observation

- Problem is equivalent to construct a good estimator for:

$$\sup_{\theta} \mathbb{E}_P f(X;\theta). \tag{7}$$

- But $\mathbb{E}\sup_\theta \frac{1}{n}\sum_{i=1}^m f(X_i;\theta) \neq \sup_\theta \mathbb{E}_P f(X;\theta)$.
- Can we construct a random variable Z_n based on $\{X_i\}$ s.t. $\mathbb{E}Z_n = \sup_\theta \mathbb{E}_P f(X;\theta)$?

Multilevel Monte-Carlo method

- Multilevel Monte-Carlo method [BGP19]: Given 2^{N+1} i.i.d. samples $\{X_i\}_{i=1}^{2^{N+1}}$ with $N \sim Geo(g)$,

Multilevel Monte-Carlo method

- Multilevel Monte-Carlo method [BGP19]: Given 2^{N+1} i.i.d. samples $\{X_i\}_{i=1}^{2^{N+1}}$ with $N \sim Geo(g)$,

$$\Delta_N = \sup_\theta \frac{1}{2^{N+1}} \sum_{i=1}^{2^{N+1}} f(X_i;\theta) - \frac{1}{2}\sup_\theta \frac{1}{2^N}\sum_{i=1}^{2^N} f(X_{2i};\theta) - \frac{1}{2}\sup_\theta \frac{1}{2^N}\sum_{i=1}^{2^N} f(X_{2i-1};\theta). \tag{8}$$

Multilevel Monte-Carlo method

- Multilevel Monte-Carlo method [BGP19]: Given 2^{N+1} i.i.d. samples $\{X_i\}_{i=1}^{2^{N+1}}$ with $N \sim Geo(g)$,

$$\Delta_N = \sup_\theta \frac{1}{2^{N+1}} \sum_{i=1}^{2^{N+1}} f(X_i;\theta) - \frac{1}{2} \sup_\theta \frac{1}{2^N} \sum_{i=1}^{2^N} f(X_{2i};\theta) - \frac{1}{2} \sup_\theta \frac{1}{2^N} \sum_{i=1}^{2^N} f(X_{2i-1};\theta). \tag{8}$$

- $\mathbb{E}[\sup_\theta f(X_1;\theta) + \Delta_N / p_N] = \sup_\theta \mathbb{E}_P f(X;\theta)$ by noticing:

Multilevel Monte-Carlo method

- Multilevel Monte-Carlo method [BGP19]: Given 2^{N+1} i.i.d. samples $\{X_i\}_{i=1}^{2^{N+1}}$ with $N \sim Geo(g)$,

$$\Delta_N = \sup_\theta \frac{1}{2^{N+1}} \sum_{i=1}^{2^{N+1}} f(X_i;\theta) - \frac{1}{2} \sup_\theta \frac{1}{2^N} \sum_{i=1}^{2^N} f(X_{2i};\theta) - \frac{1}{2} \sup_\theta \frac{1}{2^N} \sum_{i=1}^{2^N} f(X_{2i-1};\theta). \tag{8}$$

- $\mathbb{E}[\sup_\theta f(X_1;\theta) + \Delta_N/p_N] = \sup_\theta \mathbb{E}_P f(X;\theta)$ by noticing:

$$\mathbb{E}[\Delta_N/p_N] = \sum_{n=0}^{+\infty} \mathbb{E}_{P_{2^{n+1}}}[\sup_\theta f(X;\theta)] - \mathbb{E}_{P_{2^n}}[\sup_\theta f(X;\theta)] \tag{9}$$

Distributionally Robust Q-learning

- Updating rule of DR Q-learning [LBB$^+$22]:

Distributionally Robust Q-learning

- Updating rule of DR Q-learning [LBB+22]:
  - For each (s, a), sample $N \sim Geo(g)$, and 2^{N+1} samples of $s' \sim P(\cdot|s, a)$.

Distributionally Robust Q-learning

- Updating rule of DR Q-learning [LBB+22]:
  - For each (s, a), sample $N \sim Geo(g)$, and 2^{N+1} samples of $s' \sim P(\cdot|s, a)$.
  - $Q_{t+1}(s, a) = (1 - \alpha_t)Q_t(s, a) + \alpha_t \widehat{\mathcal{T}}_{\text{rob,c}} Q_t(s, a)$, where $\widehat{\mathcal{T}}_{\text{rob,c}} Q_t$ is the Multilevel MC estimator.

Distributionally Robust Q-learning

- Updating rule of DR Q-learning [LBB+22]:
  - For each (s, a), sample $N \sim Geo(g)$, and 2^{N+1} samples of $s' \sim P(\cdot|s, a)$.
  - $Q_{t+1}(s, a) = (1 - \alpha_t)Q_t(s, a) + \alpha_t \widehat{\mathcal{T}}_{\text{rob,c}} Q_t(s, a)$, where $\widehat{\mathcal{T}}_{\text{rob,c}} Q_t$ is the Multilevel MC estimator.
- Choice of parameter $g \in (1/2, 3/4)$, leading $\mathbb{E}2^{N+1} = 2g/(2g - 1)$.

Distributionally Robust Q-learning

- Updating rule of DR Q-learning [LBB+22]:
  - For each (s, a), sample $N \sim Geo(g)$, and 2^{N+1} samples of $s' \sim P(\cdot|s, a)$.
  - $Q_{t+1}(s, a) = (1 - \alpha_t)Q_t(s, a) + \alpha_t \widehat{\mathcal{T}}_{\text{rob,c}} Q_t(s, a)$, where $\widehat{\mathcal{T}}_{\text{rob,c}} Q_t$ is the Multilevel MC estimator.
- Choice of parameter $g \in (1/2, 3/4)$, leading $\mathbb{E}2^{N+1} = 2g/(2g - 1)$.
- Sample complexity [WSBZ23]: $\widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}||\mathcal{A}|}{\varepsilon^2(1-\gamma)^5\rho^4}\right)$.

Distributionally Robust Q-learning

- Updating rule of DR Q-learning [LBB+22]:
  - For each (s,a), sample $N \sim Geo(g)$, and 2^{N+1} samples of $s' \sim P(\cdot|s,a)$.
  - $Q_{t+1}(s,a) = (1-\alpha_t)Q_t(s,a) + \alpha_t \widehat{\mathcal{T}}_{\text{rob,c}} Q_t(s,a)$, where $\widehat{\mathcal{T}}_{\text{rob,c}} Q_t$ is the Multilevel MC estimator.
- Choice of parameter $g \in (1/2, 3/4)$, leading $\mathbb{E}2^{N+1} = 2g/(2g-1)$.
- Sample complexity [WSBZ23]: $\widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}||\mathcal{A}|}{\varepsilon^2(1-\gamma)^5\rho^4}\right)$.
- Limitation: unknown to computation complexity.

Alternative form

- Recall the dual form:

Alternative form

- Recall the dual form:

$$\sup_{\lambda \geq 0, \eta \in \mathbb{R}} -\lambda \sum_i P_i^* f^* \left(\frac{\eta - V_i}{\lambda} \right) - \lambda \rho + \eta.$$

Alternative form

- Recall the dual form:

$$\sup_{\lambda \geq 0, \eta \in \mathbb{R}} -\lambda \sum_i P_i^* f^* \left(\frac{\eta - V_i}{\lambda} \right) - \lambda\rho + \eta.$$

- Can we apply stochastic gradient method?

Alternative form

- Recall the dual form:

$$\sup_{\lambda \geq 0, \eta \in \mathbb{R}} -\lambda \sum_i P_i^* f^* \left(\frac{\eta - V_i}{\lambda} \right) - \lambda \rho + \eta.$$

- Can we apply stochastic gradient method?
- Sadly, stochastic gradient method will fail here. [ND16]

Alternative form

- Recall the dual form:

$$\sup_{\lambda \geq 0, \eta \in \mathbb{R}} -\lambda \sum_i P_i^* f^* \left(\frac{\eta - V_i}{\lambda} \right) - \lambda \rho + \eta.$$

- Can we apply stochastic gradient method?
- Sadly, stochastic gradient method will fail here. [ND16]
- Reason: no bounded gradient because of λ.

Alternative form

- Recall the dual form:

$$\sup_{\lambda\geq 0,\eta\in\mathbb{R}} -\lambda\sum_i P_i^* f^*\left(\frac{\eta - V_i}{\lambda}\right) - \lambda\rho + \eta.$$

- Can we apply stochastic gradient method?
- Sadly, stochastic gradient method will fail here. [ND16]
- Reason: no bounded gradient because of λ.
- Set $f(t) = (t-1)^2$, dual form:

$$\sup_{\lambda\geq 0,\eta\in\mathbb{R}} -\frac{\sum_i P_i^*(\eta - V_i)_+^2}{4\lambda} - \lambda\rho + \eta - \lambda. \tag{10}$$

Alternative form

- Recall the dual form:

$$\sup_{\lambda \geq 0, \eta \in \mathbb{R}} -\lambda \sum_i P_i^* f^* \left(\frac{\eta - V_i}{\lambda} \right) - \lambda\rho + \eta.$$

- Can we apply stochastic gradient method?
- Sadly, stochastic gradient method will fail here. [ND16]
- Reason: no bounded gradient because of λ.
- Set $f(t) = (t-1)^2$, dual form:

$$\sup_{\lambda \geq 0, \eta \in \mathbb{R}} -\frac{\sum_i P_i^* (\eta - V_i)_+^2}{4\lambda} - \lambda\rho + \eta - \lambda. \tag{10}$$

- Jointly optimizing over λ, η fails.

Alternative form

- What if fix λ?

Alternative form

- What if fix λ?
- Let's try penalty form:

$$\inf_{P} \sum_{i} P_i V_i + \lambda D_f(P \| P^*)$$

Alternative form

- What if fix λ?
- Let's try penalty form:

$$\inf_P \sum_i P_i V_i + \lambda D_f(P\|P^*)$$

- Its dual form satisfies:

$$\sup_{\eta\in\mathbb{R}} -\lambda \sum_i P_i^* f^*\left(\frac{\eta - V_i}{\lambda}\right) + \eta. \tag{11}$$

Alternative form

- What if fix λ?
- Let's try penalty form:

$$\inf_P \sum_i P_i V_i + \lambda D_f(P \| P^*)$$

- Its dual form satisfies:

$$\sup_{\eta \in \mathbb{R}} -\lambda \sum_i P_i^* f^* \left(\frac{\eta - V_i}{\lambda} \right) + \eta. \tag{11}$$

- High-level idea: small ρ (constraint form) corresponds to large λ (penalty form).

Alternative form

- What if fix λ?
- Let's try penalty form:

$$\inf_P \sum_i P_i V_i + \lambda D_f(P\|P^*)$$

- Its dual form satisfies:

$$\sup_{\eta\in\mathbb{R}} -\lambda \sum_i P_i^* f^* \left(\frac{\eta - V_i}{\lambda}\right) + \eta. \tag{11}$$

- High-level idea: small ρ (constraint form) corresponds to large λ (penalty form).
- Stochastic gradient method works here!

Penalized Robust MDPs

- How to define penalty in robust MDPs?

Penalized Robust MDPs

- How to define penalty in robust MDPs?
- Intuitively, consider a new robust Bellman operator:

$$\mathcal{T}_{\mathsf{rob,p}} V(s) = \max_a \left(R(s,a) + \gamma \inf_P \sum_{s'} P(s') V(s') + \lambda D_f(P \| P^*_{s,a}) \right) \tag{12}$$

Penalized Robust MDPs

- How to define penalty in robust MDPs?
- Intuitively, consider a new robust Bellman operator:

$$\mathcal{T}_{\mathsf{rob,p}} V(s) = \max_a \left(R(s,a) + \gamma \inf_P \sum_{s'} P(s') V(s') + \lambda D_f(P \| P^*_{s,a}) \right) \tag{12}$$

- Define the value function:

$$V^{\pi}_{\mathsf{rob,p}}(s) := \inf_P \mathbb{E}_{P,\pi} \left[\sum_{t \geq 0} \gamma^t \left(R(s_t, a_t) + \lambda \gamma D_f(P_{s_t,a_t} \| P^*_{s_t,a_t}) \right) \middle| s_0 = s \right] \tag{13}$$

Penalized Robust MDPs

- Is $V^{\pi}_{\text{rob,p}}$ well defined?

Penalized Robust MDPs

- Is $V^{\pi}_{\text{rob,p}}$ well defined?

Proposition ([YWK+23])

$\mathcal{T}_{\text{rob,p}}$ is a γ-contraction operator on value space with a fixed point $V^{*}_{\text{rob,p}}$, which satisfies $V^{*}_{\text{rob,p}} = \max_{\pi} V^{\pi}_{\text{rob,p}}$.

- Is robustness preserved?

- Is robustness preserved?

Theorem (Statistical Equivalence [YWK+23])

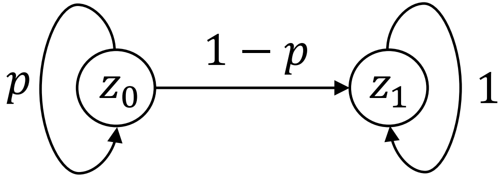
With a generative model and $f(t) = (t-1)^2$, with probability $1-\delta$,

$$\|\widehat{V}^*_{\mathsf{rob,p}} - V^*_{\mathsf{rob,p}}\|_\infty \le \varepsilon, \tag{14}$$

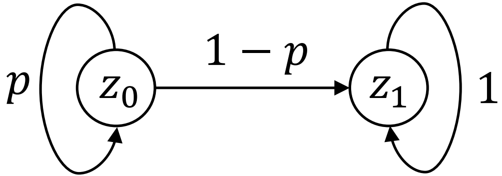
by taking $n = \widetilde{\mathcal{O}}\left(\frac{|\mathcal{S}||\mathcal{A}|}{\varepsilon^2(1-\gamma)^2}\max\{\lambda^{-2}(1-\gamma)^{-2}, \lambda^2\}\right)$. Also, there exsits a class of robust MDPs, for every (ε,δ)-correct robust RL algorithm, the total number of samples needed is at least:

$$n = \begin{cases} \widetilde{\Omega}\left(\frac{|\mathcal{S}||\mathcal{A}|\lambda^2}{\varepsilon^2(1-\gamma)^3}\right) & \text{, when } \lambda = \mathcal{O}(1-\gamma) \\ \widetilde{\Omega}\left(\frac{|\mathcal{S}||\mathcal{A}|}{\varepsilon^2(1-\gamma)^3}\right) & \text{, when } \lambda = \Omega(1-\gamma) \end{cases} \tag{15}$$

- Consider a 2-state MDP:



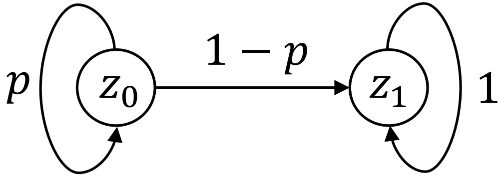
- Consider a 2-state MDP:



- Value function satisfies:

$$V(z_0) = 1 + \gamma \inf_{0 \le q \le 1} qV(z_0) + \lambda D_f(q\|p) \quad (16)$$

- Consider a 2-state MDP:

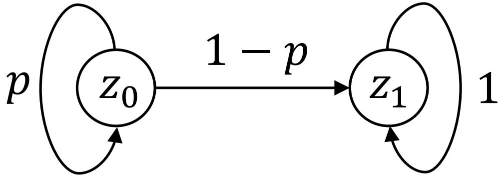


- Value function satisfies:

$$V(z_0) = 1 + \gamma \inf_{0 \le q \le 1} qV(z_0) + \lambda D_f(q\|p) \tag{16}$$

- For $f(t) = (t-1)^2$, it is a quadratic equation.

- Consider a 2-state MDP:



- Value function satisfies:

$$V(z_0) = 1 + \gamma \inf_{0 \le q \le 1} qV(z_0) + \lambda D_f(q \| p) \tag{16}$$

- For $f(t) = (t-1)^2$, it is a quadratic equation.
- Compare $V(z_0)$ under p and $p + \delta$, then apply information theory.

- We can design the robust Q-learning algorithm with a generative model:

- We can design the robust Q-learning algorithm with a generative model: At timestep t, for each (s, a), we do

- We can design the robust Q-learning algorithm with a generative model: At timestep t, for each (s,a), we do
  - Stochastic gradient method for $\sup_{\eta\in\mathbb{R}} -\lambda\sum_{s'} P^*(s'|s,a) f^*\left(\frac{\eta - V_t(s')}{\lambda}\right) + \eta$ with sufficient steps T' and obtain $\eta_{T'}(s,a)$.

- We can design the robust Q-learning algorithm with a generative model: At timestep t, for each (s,a), we do
  - Stochastic gradient method for $\sup_{\eta\in\mathbb{R}} -\lambda\sum_{s'} P^*(s'|s,a) f^*\left(\frac{\eta - V_t(s')}{\lambda}\right) + \eta$ with sufficient steps T' and obtain $\eta_{T'}(s,a)$.
  - Run one step Q-learning:

$$Q_{t+1}(s,a) = (1-\beta_t)Q_t(s,a) + \beta_t \widehat{\mathcal{T}}_{\text{rob,p}} V_t,$$

where $\widehat{\mathcal{T}}_{\text{rob,p}} V_t = r_t + \gamma \cdot \left(-\lambda f^*\left(\frac{\eta_{T'}(s,a) - V_t(s')}{\lambda}\right) + \eta_{T'}(s,a)\right).$

Robust Q-learning

- Why sufficient steps T' for inner optimization problem?

Robust Q-learning

- Why sufficient steps T' for inner optimization problem?
- Given V_t, $|\eta_{T'} - \eta^*|$ is small with high probability.

Robust Q-learning

- Why sufficient steps T' for inner optimization problem?
- Given V_t, $|\eta_{T'} - \eta^*|$ is small with high probability.
- Convergence of Q-learning requires a nearly-unbiased estimator of $\sup_{\eta\in\mathbb{R}} -\lambda \sum_{s'} P^*(s'|s,a) f^*\left(\frac{\eta - V_t(s')}{\lambda}\right) + \eta$.

Robust Q-learning

- Why sufficient steps T' for inner optimization problem?
- Given V_t, $|\eta_{T'} - \eta^*|$ is small with high probability.
- Convergence of Q-learning requires a nearly-unbiased estimator of $\sup_{\eta\in\mathbb{R}} -\lambda \sum_{s'} P^*(s'|s,a) f^*\left(\frac{\eta - V_t(s')}{\lambda}\right) + \eta$.
- With a near optimal $\eta_{T'}$, $-\lambda f^*(\frac{\eta_{T'}(s,a) - V_t(s')}{\lambda}) + \eta_{T'}(s,a)$ is nearly-unbiased.

Robust Q-learning

- Why sufficient steps T' for inner optimization problem?
- Given V_t, $|\eta_{T'} - \eta^*|$ is small with high probability.
- Convergence of Q-learning requires a nearly-unbiased estimator of $\sup_{\eta\in\mathbb{R}} -\lambda \sum_{s'} P^*(s'|s,a) f^*\left(\frac{\eta - V_t(s')}{\lambda}\right) + \eta$.
- With a near optimal $\eta_{T'}$, $-\lambda f^*(\frac{\eta_{T'}(s,a) - V_t(s')}{\lambda}) + \eta_{T'}(s,a)$ is nearly-unbiased.
- Error decomposition:

$$
\begin{aligned}
J(s'_t;\eta_{T'},V_t) - \sup_\eta \mathbb{E}J(s';\eta,V^*) &= J(s'_t;\eta_{T'},V_t) - \mathbb{E}J(s';\eta_{T'},V_t) \\
&+ \mathbb{E}J(s';\eta_{T'},V_t) - \sup_\eta \mathbb{E}J(s';\eta,V_t) \\
&+ \sup_\eta \mathbb{E}J(s';\eta,V_t) - \sup_\eta \mathbb{E}J(s';\eta,V^*) \qquad (17)
\end{aligned}
$$

Robust Q-learning

Theorem ([YWK+23])

Setting $f(t) = (t-1)^2$, $\alpha_{t'} = \frac{\lambda C}{\sqrt{t'}}$, and $\beta_t = \frac{1}{1+(1-\gamma)(t+1)}$. To obtain an ε-optimal Q-value function, the total number of sample complexity is:

$$\tilde{\mathcal{O}}\left(\frac{|\mathcal{S}||\mathcal{A}|}{\varepsilon^2(1-\gamma)^5}\right) \cdot \tilde{\mathcal{O}}\left(\frac{\max\{\lambda^2, \lambda^{-2}(1-\gamma)^{-4}\}}{\varepsilon^2(1-\gamma)^2}\right) \tag{18}$$

Limitations

- [LBB+22, WSBZ23, YWK+23] require generative model.

Limitations

- [LBB+22, WSBZ23, YWK+23] require generative model.
- Practical scenario: sampling from one trajectory.

Limitations

- [LBB+22, WSBZ23, YWK+23] require generative model.
- Practical scenario: sampling from one trajectory.
- Observed samples $(s_0, a_0, r_0, s_1, a_1, \cdots)$.

One trajectory

- [LMB+23] consider concrete cases: χ^2 and KL.

One trajectory

- [LMB$^+$23] consider concrete cases: χ^2 and KL.
- For χ^2 case, optimize over λ:

One trajectory

- [LMB+23] consider concrete cases: χ^2 and KL.
- For χ^2 case, optimize over λ:

$$\mathcal{T}_{\mathsf{rob,c}} V = R(s,a) + \gamma \sup_{\eta} \left(\eta - \sqrt{1+\rho} \sqrt{\mathbb{E}_{P^*_{s,a}} (\eta - V(s'))_+^2} \right) \tag{19}$$

One trajectory

- [LMB+23] consider concrete cases: χ^2 and KL.
- For χ^2 case, optimize over λ:

$$\mathcal{T}_{\text{rob,c}}V = R(s,a) + \gamma \sup_{\eta} \left(\eta - \sqrt{1+\rho} \sqrt{\mathbb{E}_{P^*_{s,a}} (\eta - V(s'))_+^2} \right) \tag{19}$$

- Gradient w.r.t. η:

One trajectory

- [LMB+23] consider concrete cases: χ^2 and KL.
- For χ^2 case, optimize over λ:

$$\mathcal{T}_{\text{rob,c}}V = R(s,a) + \gamma \sup_{\eta}\left(\eta - \sqrt{1+\rho}\sqrt{\mathbb{E}_{P^*_{s,a}}(\eta - V(s'))_+^2}\right) \tag{19}$$

- Gradient w.r.t. η:

$$\begin{aligned} g(\eta;V) &= 1 - \sqrt{1+\rho}\frac{\mathbb{E}_{P^*_{s,a}}(\eta - V(s'))_+}{\sqrt{\mathbb{E}_{P^*_{s,a}}(\eta - V(s'))_+^2}} \\ &= 1 - \sqrt{1+\rho}\frac{Z_1}{\sqrt{Z_2}} \end{aligned} \tag{20}$$

One trajectory

- Updating rule (at timestep t):

One trajectory

- Updating rule (at timestep t):

$$Z_{t+1,1}(s_t, a_t) = (1 - \alpha_{t,1}) Z_{t,1}(s_t, a_t) + \alpha_{t,1}(\eta_t(s_t, a_t) - V_t(s_{t+1}))_+ \quad (21)$$

$$Z_{t+1,2}(s_t, a_t) = (1 - \alpha_{t,2}) Z_{t,2}(s_t, a_t) + \alpha_{t,2}(\eta_t(s_t, a_t) - V_t(s_{t+1}))_+^2 \quad (22)$$

$$\eta_{t+1}(s_t, a_t) = (1 - \alpha_{t,3}) \eta_t(s_t, a_t) + \alpha_{t,3}(1 - \sqrt{1+\rho} \frac{Z_{t,1}(s_t, a_t)}{\sqrt{Z_{t,2}(s_t, a_t)}}) \quad (23)$$

$$\begin{aligned} Q_{t+1}(s_t, a_t) = & (1 - \alpha_{t,4}) Q_t(s_t, a_t) \\ & + \alpha_{t,4}(R(s_t, a_t) + \gamma(\eta_t(s_t, a_t) - \sqrt{1+\rho}\sqrt{Z_{t,2}(s_t, a_t)})) \end{aligned} \quad (24)$$

One trajectory

- Updating rule (at timestep t):

$$Z_{t+1,1}(s_t,a_t) = (1-\alpha_{t,1})Z_{t,1}(s_t,a_t) + \alpha_{t,1}(\eta_t(s_t,a_t) - V_t(s_{t+1}))_+ \quad (21)$$

$$Z_{t+1,2}(s_t,a_t) = (1-\alpha_{t,2})Z_{t,2}(s_t,a_t) + \alpha_{t,2}(\eta_t(s_t,a_t) - V_t(s_{t+1}))_+^2 \quad (22)$$

$$\eta_{t+1}(s_t,a_t) = (1-\alpha_{t,3})\eta_t(s_t,a_t) + \alpha_{t,3}\Big(1 - \sqrt{1+\rho}\frac{Z_{t,1}(s_t,a_t)}{\sqrt{Z_{t,2}(s_t,a_t)}}\Big) \quad (23)$$

$$Q_{t+1}(s_t,a_t) = (1-\alpha_{t,4})Q_t(s_t,a_t) + \alpha_{t,4}\big(R(s_t,a_t) + \gamma(\eta_t(s_t,a_t) - \sqrt{1+\rho}\sqrt{Z_{t,2}(s_t,a_t)})\big) \quad (24)$$

Theorem ([LMB+23])

As $t \to \infty$, $(Z_{t,1}, Z_{t,2}, \eta_t, Q_t)$ converges to $(Z_1^*, Z_2^*, \eta^*, Q^*)$ a.s.

One trajectory

- Assumption: $\alpha_{t,1} = \alpha_{t,2} = o(\alpha_{t,3})$, and $\alpha_{t,3} = o(\alpha_{t,4})$.

One trajectory

- Assumption: $\alpha_{t,1} = \alpha_{t,2} = o(\alpha_{t,3})$, and $\alpha_{t,3} = o(\alpha_{t,4})$.
- Updating Z_t with fixed η_t and Q_t:

One trajectory

- Assumption: $\alpha_{t,1} = \alpha_{t,2} = o(\alpha_{t,3})$, and $\alpha_{t,3} = o(\alpha_{t,4})$.
- Updating Z_t with fixed η_t and Q_t:

$$\begin{aligned} \dot{Z}(t) &= f(Z(t), \eta(t), Q(t)), \\ \dot{\eta}(t) &= 0, \dot{Q}(t) = 0. \end{aligned} \tag{25}$$

One trajectory

- Assumption: $\alpha_{t,1} = \alpha_{t,2} = o(\alpha_{t,3})$, and $\alpha_{t,3} = o(\alpha_{t,4})$.
- Updating Z_t with fixed η_t and Q_t:

$$\begin{aligned} \dot{Z}(t) &= f(Z(t), \eta(t), Q(t)), \\ \dot{\eta}(t) &= 0, \dot{Q}(t) = 0. \end{aligned} \tag{25}$$

$Z(t) \to \lambda_1(\eta, Q)$.

One trajectory

- Assumption: $\alpha_{t,1} = \alpha_{t,2} = o(\alpha_{t,3})$, and $\alpha_{t,3} = o(\alpha_{t,4})$.
- Updating Z_t with fixed η_t and Q_t:

$$\begin{aligned} \dot{Z}(t) &= f(Z(t), \eta(t), Q(t)), \\ \dot{\eta}(t) &= 0, \dot{Q}(t) = 0. \end{aligned} \tag{25}$$

  $Z(t) \to \lambda_1(\eta, Q)$.
- Updating η_t with fixed Q_t:

One trajectory

- Assumption: $\alpha_{t,1} = \alpha_{t,2} = o(\alpha_{t,3})$, and $\alpha_{t,3} = o(\alpha_{t,4})$.
- Updating Z_t with fixed η_t and Q_t:

$$\begin{aligned}\dot{Z}(t) &= f(Z(t), \eta(t), Q(t)), \\ \dot{\eta}(t) &= 0, \dot{Q}(t) = 0.\end{aligned} \tag{25}$$

$Z(t) \to \lambda_1(\eta, Q)$.

- Updating η_t with fixed Q_t:

$$\dot{\eta}(t) = g(\lambda_1(\eta(t), Q(t)), \eta(t), Q(t)), \dot{Q}(t) = 0. \tag{26}$$

One trajectory

- Assumption: $\alpha_{t,1} = \alpha_{t,2} = o(\alpha_{t,3})$, and $\alpha_{t,3} = o(\alpha_{t,4})$.
- Updating Z_t with fixed η_t and Q_t:

$$\begin{aligned} \dot{Z}(t) &= f(Z(t), \eta(t), Q(t)), \\ \dot{\eta}(t) &= 0, \dot{Q}(t) = 0. \end{aligned} \tag{25}$$

$Z(t) \to \lambda_1(\eta, Q)$.

- Updating η_t with fixed Q_t:

$$\dot{\eta}(t) = g(\lambda_1(\eta(t), Q(t)), \eta(t), Q(t)), \dot{Q}(t) = 0. \tag{26}$$

$\eta(t) \to \lambda_2(Q)$.

One trajectory

- Assumption: $\alpha_{t,1} = \alpha_{t,2} = o(\alpha_{t,3})$, and $\alpha_{t,3} = o(\alpha_{t,4})$.
- Updating Z_t with fixed η_t and Q_t:

$$\dot{Z}(t) = f(Z(t), \eta(t), Q(t)),$$
$$\dot{\eta}(t) = 0, \dot{Q}(t) = 0. \tag{25}$$

$Z(t) \to \lambda_1(\eta, Q)$.

- Updating η_t with fixed Q_t:

$$\dot{\eta}(t) = g(\lambda_1(\eta(t), Q(t)), \eta(t), Q(t)), \dot{Q}(t) = 0. \tag{26}$$

$\eta(t) \to \lambda_2(Q)$.

- Finally,

$$\dot{Q}(t) = h(\lambda_1(Q(t)), \lambda_2(Q(t)), Q(t)). \tag{27}$$

- 2 approaches to get a good estimator of non-linear objective:

- 2 approaches to get a good estimator of non-linear objective:
  - Multilevel Monte-Carlo

- 2 approaches to get a good estimator of non-linear objective:
  - Multilevel Monte-Carlo
  - Stochastic Gradient

- 2 approaches to get a good estimator of non-linear objective:
    - Multilevel Monte-Carlo
    - Stochastic Gradient
- Computation complexity for MMC is not considered.

- 2 approaches to get a good estimator of non-linear objective:
  - Multilevel Monte-Carlo
  - Stochastic Gradient
- Computation complexity for MMC is not considered.
- Stochastic gradient estimator needs a variant of robust MDPs, which still preserves robustness.

- 2 approaches to get a good estimator of non-linear objective:
  - Multilevel Monte-Carlo
  - Stochastic Gradient
- Computation complexity for MMC is not considered.
- Stochastic gradient estimator needs a variant of robust MDPs, which still preserves robustness.
- Data generating mechanism: from generative model to one trajectory. No finite-sample results for now.

[BGP19] Jose H Blanchet, Peter W Glynn, and Yanan Pei.
Unbiased multilevel monte carlo: Stochastic optimization, steady-state simulation, quantiles, and other applications.
arXiv preprint arXiv:1904.09929, 2019.

[LBB+22] Zijian Liu, Qinxun Bai, Jose Blanchet, Perry Dong, Wei Xu, Zhengqing Zhou, and Zhengyuan Zhou.
Distributionally robust q-learning.
In *International Conference on Machine Learning*, pages 13623–13643. PMLR, 2022.

[LMB+23] Zhipeng Liang, Xiaoteng Ma, Jose Blanchet, Jiheng Zhang, and Zhengyuan Zhou.
Single-trajectory distributionally robust reinforcement learning.
arXiv preprint arXiv:2301.11721, 2023.

[LYZJ21] Xiang Li, Wenhao Yang, Zhihua Zhang, and Michael I Jordan.
Polyak-ruppert averaged q-leaning is statistically efficient.
arXiv preprint arXiv:2112.14582, 2021.

[ND16] Hongseok Namkoong and John C Duchi.
Stochastic gradient methods for distributionally robust optimization with f-divergences.
Advances in neural information processing systems, 29, 2016.

[Wai19] Martin J Wainwright.
Stochastic approximation with cone-contractive operators: Sharp ℓ_∞-bounds for q-learning.
arXiv preprint arXiv:1905.06265, 2019.

[WSBZ23] Shengbo Wang, Nian Si, Jose Blanchet, and Zhengyuan Zhou.
A finite sample complexity bound for distributionally robust q-learning.
In *International Conference on Artificial Intelligence and Statistics*, pages 3370–3398. PMLR, 2023.

[YWK+23] Wenhao Yang, Han Wang, Tadashi Kozuno, Scott M Jordan, and Zhihua Zhang.
Avoiding model estimation in robust markov decision processes with a generative model.
arXiv preprint arXiv:2302.01248, 2023.

[YZZ22] Wenhao Yang, Liangyu Zhang, and Zhihua Zhang.
Toward theoretical understandings of robust Markov decision processes: Sample complexity and asymptotics.
The Annals of Statistics, 50(6):3223 – 3248, 2022.