Double Pessimism is Provably Efficient for Distributionally Robust Offline Reinforcement Learning

Generic Algorithm & Robust Partial Coverage Data

Miao Lu

Stanford University

INFORMS 2023

Joint work with

Han Zhong
PKU

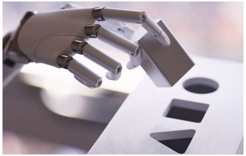
Jose Blanchet
Stanford

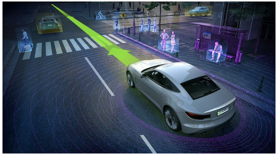
Tong Zhang
HKUST

Blanchet, J., Lu, M., Zhang, T., & Zhong, H. (2023). Double pessimism is provably efficient for distributionally robust offline reinforcement learning: Generic algorithm and robust partial coverage. *Preliminary version at 37th Conference on Neural Information Processing Systems (NeurIPS), 2023*

Offline Reinforcement Learning

Offline RL: learning optimal decisions from fixed offline datasets



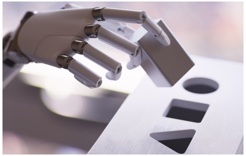


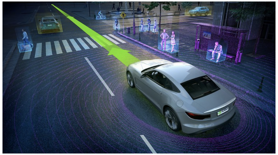
Offline RL has achieved great success in various domains, but ...

Challenge: Sim-to-Real Gap

Offline Reinforcement Learning

Offline RL: learning optimal decisions from fixed offline datasets





Offline RL has achieved great success in various domains, but ...

Challenge: Sim-to-Real Gap

Challenge: Sim-to-Real Gap

A general problem: mismatch between the dynamics of training and testing environments:

$$\mathbb{P}_{\text{Train Env.}}(\cdot) \neq \mathbb{P}_{\text{Test Env.}}(\cdot)$$

Non-robust offline RL methods will fail to generalize to testing environments :(

Solution: (distributionally) robust offline RL

- Takes the discrepancy between training and testing environments into account :)
- Seeks to find an optimal decision policy that is robust to the worst case testing environment.
- Mathematically, combines the framework of
  - Distributionally robust optimization (DRO)
  - Markov decision process (MDP)

Challenge: Sim-to-Real Gap

A general problem: mismatch between the dynamics of training and testing environments:

$$\mathbb{P}_{\text{Train Env.}}(\cdot) \neq \mathbb{P}_{\text{Test Env.}}(\cdot)$$

Non-robust offline RL methods will fail to generalize to testing environments :(

Solution: (distributionally) robust offline RL

- Takes the discrepancy between training and testing environments into account :)
- Seeks to find an optimal decision policy that is robust to the worst case testing environment.
- Mathematically, combines the framework of
  - Distributionally robust optimization (DRO)
  - Markov decision process (MDP)

A Review of Standard Offline RL

Offline RL uses the framework of **Markov decision process (MDP)**: $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, \mathbb{P}^\star, R)$.

- We consider a finite-horizon decision process.
- $\mathbb{P}^\star = \{\mathbb{P}^\star_h\}_{h\in[H]}$ and $R = \{R_h\}_{h\in[H]}$.

Interaction protocol: an agent interacts with $\mathcal{M}$ in the form of episodes (H steps). In each episode:

- at each step $h \in [H]$, the agent observes a state $s_h \in \mathcal{S}$, and takes an action $a_h \in \mathcal{A}$.
- the env. transits to $s_{h+1} \sim \mathbb{P}^\star_h(\cdot|s_h, a_h)$, and the agent receives reward $r_h = R_h(s_h, a_h)$.
- the episode ends after H decision steps.

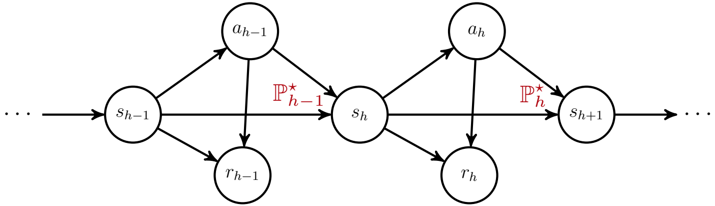
A Review of Standard Offline RL

Offline RL uses the framework of **Markov decision process (MDP)**: $\mathcal{M} = (\mathcal{S}, \mathcal{A}, H, \mathbb{P}^\star, R)$.

- We consider a finite-horizon decision process.
- $\mathbb{P}^\star = \{\mathbb{P}_h^\star\}_{h\in[H]}$ and $R = \{R_h\}_{h\in[H]}$.

Interaction protocol: an agent interacts with $\mathcal{M}$ in the form of episodes (H steps). In each episode:

- at each step $h \in [H]$, the agent observes a state $s_h \in \mathcal{S}$, and takes an action $a_h \in \mathcal{A}$.
- the env. transits to $s_{h+1} \sim \mathbb{P}_h^\star(\cdot|s_h, a_h)$, and the agent receives reward $r_h = R_h(s_h, a_h)$.
- the episode ends after H decision steps.



A Review of Standard Offline RL

Goal of offline RL: given an offline dataset $\mathcal{D}$ with N trajectories (episodes):

$$\mathcal{D} = \left\{ \left(s_h^\tau, a_h^\tau, r_h^\tau, s_{h+1}^\tau \right) \right\}_{h \in [H], \tau \in [N]} \quad a_h^\tau \sim \pi_h^{\mathrm{b}}(\cdot | s_h^\tau), \quad s_{h+1}^\tau \sim \mathbb{P}_h^\star(\cdot | s_h^\tau, a_h^\tau)$$

find the optimal policy $\pi^\star = \{\pi_h^\star : \mathcal{S} \mapsto \mathcal{A}\}_{h \in [H]}$ that maximizes expected total reward:

$$\pi^\star \in \operatorname*{arg\,max}_{\pi = \{\pi_h\}_{h \in [H]} : \pi_h : \mathcal{S} \mapsto \mathcal{A}} V_1^\pi(s_1; \mathbb{P}^\star)$$

- The total reward of π from step h:

$$V_h^\pi(s_h; \mathbb{P}^\star) := \mathbb{E}_{\pi, \mathbb{P}^\star} \left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) \middle| s_h; \ a_{h'} \sim \pi_{h'}(\cdot | s_{h'}), s_{h'+1} \sim \mathbb{P}_{h'}^\star(\cdot | s_{h'}, a_{h'}) \right]$$

- No interaction with the real environment, only have $\mathcal{D}$.
- **The policy is evaluated on the same dynamics $\mathbb{P}^\star$ as the data generation process!**

A Review of Standard Offline RL

Goal of offline RL: given an offline dataset $\mathcal{D}$ with N trajectories (episodes):

$$\mathcal{D} = \left\{ \left(s_h^\tau, a_h^\tau, r_h^\tau, s_{h+1}^\tau \right) \right\}_{h\in[H],\tau\in[N]} \quad a_h^\tau \sim \pi_h^{\mathrm{b}}(\cdot|s_h^\tau), \quad s_{h+1}^\tau \sim \mathbb{P}_h^\star(\cdot|s_h^\tau, a_h^\tau)$$

find the optimal policy $\pi^\star = \{\pi_h^\star : \mathcal{S} \mapsto \mathcal{A}\}_{h\in[H]}$ that maximizes expected total reward:

$$\pi^\star \in \mathop{\arg\max}_{\pi=\{\pi_h\}_{h\in[H]}:\pi_h:\mathcal{S}\mapsto\mathcal{A}} V_1^\pi(s_1; \mathbb{P}^\star)$$

- The total reward of π from step h:

$$V_h^\pi(s_h; \mathbb{P}^\star) := \mathbb{E}_{\pi,\mathbb{P}^\star}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) \middle| s_h;\; a_{h'} \sim \pi_{h'}(\cdot|s_{h'}), s_{h'+1} \sim \mathbb{P}_{h'}^\star(\cdot|s_{h'}, a_{h'}) \right]$$

- No interaction with the real environment, only have $\mathcal{D}$.
- **The policy is evaluated on the same dynamics $\mathbb{P}^\star$ as the data generation process!**

A Unified Framework of Robust Offline RL

Robust offline RL considers the discrepancy between training and testing environments.

It uses the framework of **robust Markov decision process (RMDP)**:

$$\mathcal{M}_{\boldsymbol{\Phi}} = (\mathcal{S}, \mathcal{A}, H, \mathbb{P}^\star, R, \boldsymbol{\Phi})$$

- $\boldsymbol{\Phi}$ denotes the robust set of transition dynamics,
- Interpretations of $\mathbb{P}^\star$ and $\boldsymbol{\Phi}$:
  - $\mathbb{P}^\star$: the dynamic of the training environment (nominal transition kernel).
  - $\mathbb{P}' \in \boldsymbol{\Phi}$: a possible dynamic of the testing environments.
- Ususally, $\boldsymbol{\Phi}$ is a "ball of distribution" centered at $\mathbb{P}^\star$.
  - e.g., ϕ-divergence ball, wasserstein ball.

A Unified Framework of Robust Offline RL

Robust offline RL considers the discrepancy between training and testing environments.

It uses the framework of **robust Markov decision process (RMDP)**:

$$\mathcal{M}_{\boldsymbol{\Phi}} = (\mathcal{S}, \mathcal{A}, H, \mathbb{P}^{\star}, R, \boldsymbol{\Phi})$$

- $\boldsymbol{\Phi}$ denotes the robust set of transition dynamics,
- Interpretations of $\mathbb{P}^{\star}$ and $\boldsymbol{\Phi}$:
  - $\mathbb{P}^{\star}$: the dynamic of the training environment (nominal transition kernel).
  - $\mathbb{P}' \in \boldsymbol{\Phi}$: a possible dynamic of the testing environments.
- Ususally, $\boldsymbol{\Phi}$ is a "ball of distribution" centered at $\mathbb{P}^{\star}$.
  - e.g., ϕ-divergence ball, wasserstein ball.

A Unified Framework of Robust Offline RL

Robust offline RL considers the discrepancy between training and testing environments.

It uses the framework of **robust Markov decision process (RMDP)**:

$$\mathcal{M}_{\boldsymbol{\Phi}} = (\mathcal{S}, \mathcal{A}, H, \mathbb{P}^\star, R, \boldsymbol{\Phi})$$

- $\boldsymbol{\Phi}$ denotes the robust set of transition dynamics,
- Interpretations of $\mathbb{P}^\star$ and $\boldsymbol{\Phi}$:
  - $\mathbb{P}^\star$: the dynamic of the training environment (nominal transition kernel).
  - $\mathbb{P}' \in \boldsymbol{\Phi}$: a possible dynamic of the testing environments.
- Ususally, $\boldsymbol{\Phi}$ is a "ball of distribution" centered at $\mathbb{P}^\star$.
  - e.g., ϕ-divergence ball, wasserstein ball.

A Unified Framework of Robust Offline RL

Goal of robust offline RL: given an offline dataset collected from environment $\mathbb{P}^\star$:

$$\mathcal{D} = \Big\{ \big(s_h^\tau, a_h^\tau, r_h^\tau, s_{h+1}^\tau \big) \Big\}_{h \in [H], \tau \in [N]} \quad a_h^\tau \sim \pi_h^{\mathrm{b}}(\cdot | s_h^\tau), \quad s_{h+1}^\tau \sim \mathbb{P}_h^\star(\cdot | s_h^\tau, a_h^\tau)$$

find the optimal robust policy $\pi^\star = \{\pi_h^\star : \mathcal{S} \mapsto \mathcal{A}\}_{h \in [H]}$ that maximizes the robust expected total rewards:

$$\pi^\star \in \mathop{\arg\max}_{\pi = \{\pi_h\}_{h \in [H]} : \pi_h : \mathcal{S} \mapsto \mathcal{A}} \ \min_{\mathbb{P}' = \{\mathbb{P}_h'\}_{h \in [H]} : \mathbb{P}_h' \in \boldsymbol{\Phi}} V_1^\pi(s_1; \mathbb{P}')$$

- $V_1^\pi(s_h; \mathbb{P}')$ is same defined as in MDP, but now $\pi^\star$ maximizes the worst case value.
- No access to data from $\mathbb{P}' \in \boldsymbol{\Phi}$. Only have $\mathcal{D}$ collected from $\mathbb{P}^\star$.

The policy is evaluated on the worst case dynamics $\mathbb{P}' \in \boldsymbol{\Phi}$ of the testing environments!

A Unified Framework of Robust Offline RL

Goal of robust offline RL: given an offline dataset collected from environment $\mathbb{P}^\star$:

$$\mathcal{D} = \Big\{\big(s_h^\tau, a_h^\tau, r_h^\tau, s_{h+1}^\tau\big)\Big\}_{h\in[H],\tau\in[N]} \quad a_h^\tau \sim \pi_h^{\mathrm{b}}(\cdot|s_h^\tau), \quad s_{h+1}^\tau \sim \mathbb{P}_h^\star(\cdot|s_h^\tau, a_h^\tau)$$

find the optimal robust policy $\pi^\star = \{\pi_h^\star : \mathcal{S} \mapsto \mathcal{A}\}_{h\in[H]}$ that maximizes the robust expected total rewards:

$$\pi^\star \in \mathop{\arg\max}_{\pi=\{\pi_h\}_{h\in[H]}:\pi_h:\mathcal{S}\mapsto\mathcal{A}} \ \min_{\mathbb{P}'=\{\mathbb{P}'_h\}_{h\in[H]}:\mathbb{P}'_h\in\boldsymbol{\Phi}} V_1^\pi(s_1;\mathbb{P}')$$

- $V_1^\pi(s_h;\mathbb{P}')$ is same defined as in MDP, but now $\pi^\star$ maximizes the worst case value.
- No access to data from $\mathbb{P}' \in \boldsymbol{\Phi}$. Only have $\mathcal{D}$ collected from $\mathbb{P}^\star$.

The policy is evaluated on the worst case dynamics $\mathbb{P}' \in \boldsymbol{\Phi}$ of the testing environments!

A Unified Framework of Robust Offline RL

Goal of robust offline RL: given an offline dataset collected from environment $\mathbb{P}^\star$:

$$\mathcal{D} = \Big\{ \big(s_h^\tau, a_h^\tau, r_h^\tau, s_{h+1}^\tau \big) \Big\}_{h \in [H], \tau \in [N]} \quad a_h^\tau \sim \pi_h^{\mathrm{b}}(\cdot | s_h^\tau), \quad s_{h+1}^\tau \sim \mathbb{P}_h^\star(\cdot | s_h^\tau, a_h^\tau)$$

find the optimal robust policy $\pi^\star = \{\pi_h^\star : \mathcal{S} \mapsto \mathcal{A}\}_{h \in [H]}$ that maximizes the robust expected total rewards:

$$\pi^\star \in \mathop{\arg\max}_{\pi = \{\pi_h\}_{h \in [H]} : \pi_h : \mathcal{S} \mapsto \mathcal{A}} \ \min_{\mathbb{P}' = \{\mathbb{P}'_h\}_{h \in [H]} : \mathbb{P}'_h \in \mathbf{\Phi}} V_1^\pi(s_1; \mathbb{P}')$$

- $V_1^\pi(s_h; \mathbb{P}')$ is same defined as in MDP, but now $\pi^\star$ maximizes the worst case value.
- No access to data from $\mathbb{P}' \in \mathbf{\Phi}$. Only have $\mathcal{D}$ collected from $\mathbb{P}^\star$.

The policy is evaluated on the worst case dynamics $\mathbb{P}' \in \mathbf{\Phi}$ of the testing environments!

Questions:

Q1: Is there a principled way to obtain optimal sample efficiency for robust offline RL under minimal data assumptions?

A1: Yes! "Double pessimism" is the answer.

Q2: Can this principle lead to a generic algorithm in the context of large state space and function approximation?

A2: Yes! Our algorithm "Doubly Pessimistic Model-based Policy Optimization" (P^2MPO).

Questions:

Q1: Is there a principled way to obtain optimal sample efficiency for robust offline RL under minimal data assumptions?

A1: Yes! "Double pessimism" is the answer.

Q2: Can this principle lead to a generic algorithm in the context of large state space and function approximation?

A2: Yes! Our algorithm "Doubly Pessimistic Model-based Policy Optimization" (P^2MPO).

More Detailed Setups

- For simplicity, we assume that the reward function R is known to the learner.
- Robust mapping $\mathbf{\Phi} : \mathcal{P} \mapsto 2^{\mathcal{P}}$, with $\mathcal{P} := \{\mathbb{P}_h(\cdot|\cdot,\cdot) : \mathcal{S} \times \mathcal{A} \mapsto \Delta(\mathcal{S})\}$.
  - $\mathbf{\Phi}(\mathbb{P}_h)$ is the robust set of $\mathbb{P}_h \in \mathcal{P}$.
- Robust value functions: we define for each $\mathbb{P} = \{\mathbb{P}_h\}_{h\in[H]} \subset \mathcal{P}$ and policy π,

$$V_{h,\mathbb{P},\mathbf{\Phi}}^{\pi}(s) := \min_{\substack{\mathbb{P}'_h \in \mathbf{\Phi}(\mathbb{P}_h) \\ 1\le h\le H}} \mathbb{E}_{\pi,\mathbb{P}'}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) \middle| s_h; a_{h'} \sim \pi_{h'}(\cdot|s_{h'}), s_{h'+1} \sim \mathbb{P}'_{h'}(\cdot|s_{h'}, a_{h'})\right],$$

- Formally, the goal is to find a policy $\widehat{\pi}$ from $\mathcal{D}$ that minimizes its suboptimality gap from $\pi^\star$:

$$\mathrm{SubOpt}(\widehat{\pi}; s_1) := V_{1,\mathbb{P},\mathbf{\Phi}}^{\pi^\star}(s_1) - V_{1,\mathbb{P},\mathbf{\Phi}}^{\widehat{\pi}}(s_1),$$

Here $\pi^\star$ is the optimal robust policy (for simplicity, we assume a fixed $s_1 \in \mathcal{S}$).

More Detailed Setups

- For simplicity, we assume that the reward function R is known to the learner.
- Robust mapping $\mathbf{\Phi} : \mathcal{P} \mapsto 2^{\mathcal{P}}$, with $\mathcal{P} := \{\mathbb{P}_h(\cdot|\cdot,\cdot) : \mathcal{S} \times \mathcal{A} \mapsto \Delta(\mathcal{S})\}$.
  - $\mathbf{\Phi}(\mathbb{P}_h)$ is the robust set of $\mathbb{P}_h \in \mathcal{P}$.
- Robust value functions: we define for each $\mathbb{P} = \{\mathbb{P}_h\}_{h\in[H]} \subset \mathcal{P}$ and policy π,

$$V_{h,\mathbb{P},\mathbf{\Phi}}^{\pi}(s) := \min_{\substack{\mathbb{P}'_h \in \mathbf{\Phi}(\mathbb{P}_h) \\ 1\leq h\leq H}} \mathbb{E}_{\pi,\mathbb{P}'}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) \middle| s_h; a_{h'} \sim \pi_{h'}(\cdot|s_{h'}), s_{h'+1} \sim \mathbb{P}'_{h'}(\cdot|s_{h'}, a_{h'})\right],$$

- Formally, the goal is to find a policy $\widehat{\pi}$ from $\mathcal{D}$ that minimizes its suboptimality gap from $\pi^\star$:

$$\mathrm{SubOpt}(\widehat{\pi}; s_1) := V_{1,\mathbb{P},\mathbf{\Phi}}^{\pi^\star}(s_1) - V_{1,\mathbb{P},\mathbf{\Phi}}^{\widehat{\pi}}(s_1),$$

Here $\pi^\star$ is the optimal robust policy (for simplicity, we assume a fixed $s_1 \in \mathcal{S}$).

More Detailed Setups

- For simplicity, we assume that the reward function R is known to the learner.
- Robust mapping $\mathbf{\Phi} : \mathcal{P} \mapsto 2^{\mathcal{P}}$, with $\mathcal{P} := \{\mathbb{P}_h(\cdot|\cdot,\cdot) : \mathcal{S} \times \mathcal{A} \mapsto \Delta(\mathcal{S})\}$.
  - $\mathbf{\Phi}(\mathbb{P}_h)$ is the robust set of $\mathbb{P}_h \in \mathcal{P}$.
- Robust value functions: we define for each $\mathbb{P} = \{\mathbb{P}_h\}_{h\in[H]} \subset \mathcal{P}$ and policy π,

$$V_{h,\mathbb{P},\mathbf{\Phi}}^{\pi}(s) := \min_{\substack{\mathbb{P}'_h \in \mathbf{\Phi}(\mathbb{P}_h) \\ 1\le h\le H}} \mathbb{E}_{\pi,\mathbb{P}'}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'},a_{h'}) \middle| s_h; a_{h'} \sim \pi_{h'}(\cdot|s_{h'}), s_{h'+1} \sim \mathbb{P}'_{h'}(\cdot|s_{h'},a_{h'})\right],$$

- Formally, the goal is to find a policy $\widehat{\pi}$ from $\mathcal{D}$ that minimizes its suboptimality gap from $\pi^\star$:

$$\mathrm{SubOpt}(\widehat{\pi}; s_1) := V_{1,\mathbb{P},\mathbf{\Phi}}^{\pi^\star}(s_1) - V_{1,\mathbb{P},\mathbf{\Phi}}^{\widehat{\pi}}(s_1),$$

Here $\pi^\star$ is the optimal robust policy (for simplicity, we assume a fixed $s_1 \in \mathcal{S}$).

More Detailed Setups

- For simplicity, we assume that the reward function R is known to the learner.
- Robust mapping $\mathbf{\Phi} : \mathcal{P} \mapsto 2^{\mathcal{P}}$, with $\mathcal{P} := \{\mathbb{P}_h(\cdot|\cdot,\cdot) : \mathcal{S} \times \mathcal{A} \mapsto \Delta(\mathcal{S})\}$.
  - $\mathbf{\Phi}(\mathbb{P}_h)$ is the robust set of $\mathbb{P}_h \in \mathcal{P}$.
- Robust value functions: we define for each $\mathbb{P} = \{\mathbb{P}_h\}_{h\in[H]} \subset \mathcal{P}$ and policy π,

$$V_{h,\mathbb{P},\mathbf{\Phi}}^{\pi}(s) := \min_{\substack{\mathbb{P}'_h \in \mathbf{\Phi}(\mathbb{P}_h) \\ 1 \le h \le H}} \mathbb{E}_{\pi,\mathbb{P}'}\left[\sum_{h'=h}^{H} R_{h'}(s_{h'}, a_{h'}) \middle| s_h; a_{h'} \sim \pi_{h'}(\cdot|s_{h'}), s_{h'+1} \sim \mathbb{P}'_{h'}(\cdot|s_{h'}, a_{h'})\right],$$

- Formally, the goal is to find a policy $\widehat{\pi}$ from $\mathcal{D}$ that minimizes its suboptimality gap from $\pi^\star$:

$$\mathrm{SubOpt}(\widehat{\pi}; s_1) := V_{1,\mathbb{P},\mathbf{\Phi}}^{\pi^\star}(s_1) - V_{1,\mathbb{P},\mathbf{\Phi}}^{\widehat{\pi}}(s_1),$$

Here $\pi^\star$ is the optimal robust policy (for simplicity, we assume a fixed $s_1 \in \mathcal{S}$).

Main Challenges

Distributional shifts from two souces:

- The mismatch between the training environment $\mathbb{P}^\star_h$ and the testing environment $\mathbb{P}' \in \mathbf{\Phi}(\mathbb{P}^\star_h)$.
  - we only have data from $\mathbb{P}^\star$, but we need to evaluate on distributions induced by $\mathbb{P}'$.
- The mismatch between the behavior policy π^{b} and the target policies $\widehat{\pi}$ to be learned.
  - we only have data from π^{b}, but we need to evaluate on distributions induced by learned $\widehat{\pi}$.

Large state space $\mathcal{S}$:

- The state space can be infinite in general, where existing methods for tabular RMDPs fail.

Main Challenges

Distributional shifts from two souces:

- The mismatch between the training environment $\mathbb{P}_h^\star$ and the testing environment $\mathbb{P}' \in \mathbf{\Phi}(\mathbb{P}_h^\star)$.
  - we only have data from $\mathbb{P}^\star$, but we need to evaluate on distributions induced by $\mathbb{P}'$.
- The mismatch between the behavior policy π^{b} and the target policies $\widehat{\pi}$ to be learned.
  - we only have data from π^{b}, but we need to evaluate on distributions induced by learned $\widehat{\pi}$.

Large state space $\mathcal{S}$:

- The state space can be infinite in general, where existing methods for tabular RMDPs fail.

Main Challenges

Distributional shifts from two souces:

- The mismatch between the training environment $\mathbb{P}_h^\star$ and the testing environment $\mathbb{P}' \in \mathbf{\Phi}(\mathbb{P}_h^\star)$.
  - we only have data from $\mathbb{P}^\star$, but we need to evaluate on distributions induced by $\mathbb{P}'$.
- The mismatch between the behavior policy π^{b} and the target policies $\widehat{\pi}$ to be learned.
  - we only have data from π^{b}, but we need to evaluate on distributions induced by learned $\widehat{\pi}$.

Large state space $\mathcal{S}$:

- The state space can be infinite in general, where existing methods for tabular RMDPs fail.

Main Challenges

Distributional shifts from two souces:

- The mismatch between the training environment $\mathbb{P}_h^\star$ and the testing environment $\mathbb{P}' \in \mathbf{\Phi}(\mathbb{P}_h^\star)$.
  - we only have data from $\mathbb{P}^\star$, but we need to evaluate on distributions induced by $\mathbb{P}'$.
- The mismatch between the behavior policy π^{b} and the target policies $\widehat{\pi}$ to be learned.
  - we only have data from π^{b}, but we need to evaluate on distributions induced by learned $\widehat{\pi}$.

Large state space $\mathcal{S}$:

- The state space can be infinite in general, where existing methods for tabular RMDPs fail.

Pessimism: Handling Distributional Shifts

In **standard offline RL**, we have one source of distributional shift:

- The mismatch between the behavior policy π^{b} and the target policies $\widehat{\pi}$ to be learned.
- A naive attempt would require the data to cover the distributions induced by all possible policy $\widehat{\pi}$.
- The solution: being "pessimism" in the face of data uncertainty that originates from the statistical estimation of the transition kernel $\mathbb{P}^{\star}$ [Jin et al., 2021, Uehara and Sun, 2021].
- With pessimism, one can efficiently learn the optimal policy with only "partial coverage data" – only covering the trajectories induced by the optimal policy $\pi^{\star}$ (the mininal assumption).

Pessimism: Handling Distributional Shifts

In **standard offline RL**, we have one source of distributional shift:

- The mismatch between the behavior policy π^{b} and the target policies $\widehat{\pi}$ to be learned.
- A naive attempt would require the data to cover the distributions induced by all possible policy $\widehat{\pi}$.
- The solution: being "pessimism" in the face of data uncertainty that originates from the statistical estimation of the transition kernel $\mathbb{P}^{\star}$ [Jin et al., 2021, Uehara and Sun, 2021].
- With pessimism, one can efficiently learn the optimal policy with only "partial coverage data" – only covering the trajectories induced by the optimal policy $\pi^{\star}$ (the mininal assumption).

Pessimism: Handling Distributional Shifts

In **standard offline RL**, we have one source of distributional shift:

- The mismatch between the behavior policy π^{b} and the target policies $\widehat{\pi}$ to be learned.
- A naive attempt would require the data to cover the distributions induced by all possible policy $\widehat{\pi}$.
- The solution: being "pessimism" in the face of data uncertainty that originates from the statistical estimation of the transition kernel $\mathbb{P}^{\star}$ [Jin et al., 2021, Uehara and Sun, 2021].
- With pessimism, one can efficiently learn the optimal policy with only "partial coverage data" – only covering the trajectories induced by the optimal policy $\pi^{\star}$ (the mininal assumption).

Pessimism: Handling Distributional Shifts

In **standard offline RL**, we have one source of distributional shift:

- The mismatch between the behavior policy π^{b} and the target policies $\widehat{\pi}$ to be learned.
- A naive attempt would require the data to cover the distributions induced by all possible policy $\widehat{\pi}$.
- The solution: being "pessimism" in the face of data uncertainty that originates from the statistical estimation of the transition kernel $\mathbb{P}^{\star}$ [Jin et al., 2021, Uehara and Sun, 2021].
- With pessimism, one can efficiently learn the optimal policy with only "partial coverage data" – only covering the trajectories induced by the optimal policy $\pi^{\star}$ (the mininal assumption).

Double Pessimism: Handling Coupled Distributional Shifts

In **robust offline RL**, we have two coupled sources of distributional shift ($\mathbb{P}^\star$ vs $\mathbb{P}' \in \mathbf{\Phi}$, and π^{b} vs $\widehat{\pi}$).

- Solution: "double pessimism"
  - pessimism in the face of data uncertainty which originates from statistical estimation of the nominal transition kernel $\mathbb{P}^\star$;
  - pessimism in the face of testing env. uncertainty which comes from the target of finding a robust policy against the worst case testing env. $\mathbb{P}' \in \mathbf{\Phi}(\mathbb{P}^\star)$.
- However, $\mathbf{\Phi}(\mathbb{P})$ relies on $\mathbb{P}$
  - the double pessimism is coupled.
  - perform pessimism in an iterated manner!

Double Pessimism: Handling Coupled Distributional Shifts

In **robust offline RL**, we have two coupled sources of distributional shift ($\mathbb{P}^\star$ vs $\mathbb{P}' \in \boldsymbol{\Phi}$, and π^{b} vs $\widehat{\pi}$).

- Solution: "double pessimism"
  - pessimism in the face of data uncertainty which originates from statistical estimation of the nominal transition kernel $\mathbb{P}^\star$;
  - pessimism in the face of testing env. uncertainty which comes from the target of finding a robust policy against the worst case testing env. $\mathbb{P}' \in \boldsymbol{\Phi}(\mathbb{P}^\star)$.
- However, $\boldsymbol{\Phi}(\mathbb{P})$ relies on $\mathbb{P}$
  - the double pessimism is coupled.
  - perform pessimism in an iterated manner!

Double Pessimism: Handling Coupled Distributional Shifts

In **robust offline RL**, we have two coupled sources of distributional shift ($\mathbb{P}^\star$ vs $\mathbb{P}' \in \mathbf{\Phi}$, and π^{b} vs $\widehat{\pi}$).

- Solution: "double pessimism"
  - pessimism in the face of data uncertainty which originates from statistical estimation of the nominal transition kernel $\mathbb{P}^\star$;
  - pessimism in the face of testing env. uncertainty which comes from the target of finding a robust policy against the worst case testing env. $\mathbb{P}' \in \mathbf{\Phi}(\mathbb{P}^\star)$.
- However, $\mathbf{\Phi}(\mathbb{P})$ relies on $\mathbb{P}$
  - the double pessimism is coupled.
  - perform pessimism in an iterated manner!

Double Pessimism: Handling Coupled Distributional Shifts

In **robust offline RL**, we have two coupled sources of distributional shift ($\mathbb{P}^\star$ vs $\mathbb{P}' \in \mathbf{\Phi}$, and π^{b} vs $\widehat{\pi}$).

- Solution: "double pessimism"
  - pessimism in the face of data uncertainty which originates from statistical estimation of the nominal transition kernel $\mathbb{P}^\star$;
  - pessimism in the face of testing env. uncertainty which comes from the target of finding a robust policy against the worst case testing env. $\mathbb{P}' \in \mathbf{\Phi}(\mathbb{P}^\star)$.
- However, $\mathbf{\Phi}(\mathbb{P})$ relies on $\mathbb{P}$
  - the double pessimism is coupled.
  - perform pessimism in an iterated manner!

Algorithm Framework: P^2MPO

Algorithm 1: Doubly Pessimistic Model-based Policy Optimization (P^2MPO)

1. Model estimation step:

Obtain a confidence region $\widehat{\mathcal{P}} = \texttt{ModelEst}(\mathcal{D}, \mathcal{P})$ of $\mathbb{P}^\star$.

2. Doubly pessimistic policy optimization step:

Set the policy $\widehat{\pi}$ as

$$\widehat{\pi} = \arg\max_{\pi} J_{\mathtt{Pess}^2}(\pi)$$

where $J_{\mathtt{Pess}^2}(\pi)$ is defined as a doubly pessimistic value estimator:

$$J_{\mathtt{Pess}^2}(\pi) := \min_{\substack{\mathbb{P}_h \in \widehat{\mathcal{P}}_h \\ 1 \le h \le H}} \min_{\substack{\mathbb{P}'_h \in \Phi(\mathbb{P}_h) \\ 1 \le h \le H}} V_1^\pi(s_1; \mathbb{P}')$$

- One can realize P^2MPO by specifying the subalgorithm $\texttt{ModelEst}(\mathcal{D}, \mathcal{P})$ for concrete RMDPs.

Algorithm Framework: P^{2}MPO

Algorithm 1: Doubly Pessimistic Model-based Policy Optimization (P^{2}MPO)

1. **Model estimation step:**
   Obtain a confidence region $\widehat{\mathcal{P}} = \texttt{ModelEst}(\mathcal{D}, \mathcal{P})$ of $\mathbb{P}^\star$.

2. **Doubly pessimistic policy optimization step:**
   Set the policy $\widehat{\pi}$ as

$$\widehat{\pi} = \arg\max_{\pi} J_{\texttt{Pess}^2}(\pi)$$

   where $J_{\texttt{Pess}^2}(\pi)$ is defined as a doubly pessimistic value estimator:

$$J_{\texttt{Pess}^2}(\pi) := \min_{\substack{\mathbb{P}_h \in \widehat{\mathcal{P}}_h \\ 1 \le h \le H}} \min_{\substack{\mathbb{P}'_h \in \Phi(\mathbb{P}_h) \\ 1 \le h \le H}} V_1^\pi(s_1; \mathbb{P}')$$

- One can realize P^{2}MPO by specifying the subalgorithm $\texttt{ModelEst}(\mathcal{D}, \mathcal{P})$ for concrete RMDPs.

Algorithm Framework: P^{2}MPO

Algorithm 1: Doubly Pessimistic Model-based Policy Optimization (P^{2}MPO)

1. **Model estimation step:**
   Obtain a confidence region $\widehat{\mathcal{P}} = \texttt{ModelEst}(\mathcal{D}, \mathcal{P})$ of $\mathbb{P}^\star$.

2. **Doubly pessimistic policy optimization step:**
   Set the policy $\widehat{\pi}$ as

$$\widehat{\pi} = \arg\max_{\pi} J_{\texttt{Pess}^2}(\pi)$$

where $J_{\texttt{Pess}^2}(\pi)$ is defined as a doubly pessimistic value estimator:

$$J_{\texttt{Pess}^2}(\pi) := \min_{\substack{\mathbb{P}_h \in \widehat{\mathcal{P}}_h \\ 1 \le h \le H}} \min_{\substack{\mathbb{P}'_h \in \mathbf{\Phi}(\mathbb{P}_h) \\ 1 \le h \le H}} V_1^\pi(s_1; \mathbb{P}')$$

- One can realize P^{2}MPO by specifying the subalgorithm $\texttt{ModelEst}(\mathcal{D}, \mathcal{P})$ for concrete RMDPs.

Algorithm Framework: P^{2}MPO

Algorithm 1: Doubly Pessimistic Model-based Policy Optimization (P^{2}MPO)

1. **Model estimation step:**
   Obtain a confidence region $\widehat{\mathcal{P}} = \texttt{ModelEst}(\mathcal{D}, \mathcal{P})$ of $\mathbb{P}^\star$.

2. **Doubly pessimistic policy optimization step:**
   Set the policy $\widehat{\pi}$ as

$$\widehat{\pi} = \arg\max_{\pi} J_{\texttt{Pess}^2}(\pi)$$

   where $J_{\texttt{Pess}^2}(\pi)$ is defined as a doubly pessimistic value estimator:

$$J_{\texttt{Pess}^2}(\pi) := \min_{\substack{\mathbb{P}_h \in \widehat{\mathcal{P}}_h \\ 1 \le h \le H}} \min_{\substack{\mathbb{P}'_h \in \mathbf{\Phi}(\mathbb{P}_h) \\ 1 \le h \le H}} V_1^\pi(s_1; \mathbb{P}')$$

- One can realize P^{2}MPO by specifying the subalgorithm $\texttt{ModelEst}(\mathcal{D}, \mathcal{P})$ for concrete RMDPs.

Main Assumption: Robust Partial Coverage Data

$d^{\pi}_{\mathbb{P},h}(s,a)$: the state-action visitation measure at step h induced by policy π in dynamic $\mathbb{P}$.

Assumption 1 (Robust partial coverage data).

We assume that the following robust partial coverage coefficient is finite:

$$C^{\star}_{\mathbb{P}^{\star},\boldsymbol{\Phi}} := \max_{1\leq h\leq H} \max_{\substack{\mathbb{P}_h\in\boldsymbol{\Phi}(\mathbb{P}^{\star}_h)\\ 1\leq h\leq H}} \mathbb{E}_{(s,a)\sim d^{\pi^{\mathrm{b}}}_{\mathbb{P}^{\star},h}}\left[\left(\frac{d^{\pi^{\star}}_{\mathbb{P},h}(s,a)}{d^{\pi^{\mathrm{b}}}_{\mathbb{P}^{\star},h}(s,a)}\right)^2\right] < +\infty, \tag{1}$$

- This only requires that the offline data $d^{\pi^{\mathrm{b}}}_{\mathbb{P}^{\star}}$ can cover the trajectories induced by the optimal robust policy $d^{\pi^{\star}}_{\mathbb{P}}$ (for each $\mathbb{P}\in\boldsymbol{\Phi}(\mathbb{P}^{\star})$).
- Weaker and more practical than offline data from generative model or uniformly lower bounded distribution over (s,a).

Main Assumption: Robust Partial Coverage Data

$d^{\pi}_{\mathbb{P},h}(s,a)$: the state-action visitation measure at step h induced by policy π in dynamic $\mathbb{P}$.

Assumption 1 (Robust partial coverage data).

We assume that the following robust partial coverage coefficient is finite:

$$C^{\star}_{\mathbb{P}^\star,\mathbf{\Phi}} := \max_{1\leq h\leq H} \max_{\substack{\mathbb{P}_h\in\mathbf{\Phi}(\mathbb{P}^\star_h)\\1\leq h\leq H}} \mathbb{E}_{(s,a)\sim d^{\pi^{\mathrm{b}}}_{\mathbb{P}^\star,h}}\left[\left(\frac{d^{\pi^\star}_{\mathbb{P},h}(s,a)}{d^{\pi^{\mathrm{b}}}_{\mathbb{P}^\star,h}(s,a)}\right)^2\right] < +\infty, \tag{1}$$

- This only requires that the offline data $d^{\pi^{\mathrm{b}}}_{\mathbb{P}^\star}$ can cover the trajectories induced by the optimal robust policy $d^{\pi^\star}_{\mathbb{P}}$ (for each $\mathbb{P}\in\mathbf{\Phi}(\mathbb{P}^\star)$).
- Weaker and more practical than offline data from generative model or uniformly lower bounded distribution over (s,a).

Main Assumption: Robust Partial Coverage Data

$d^{\pi}_{\mathbb{P},h}(s,a)$: the state-action visitation measure at step h induced by policy π in dynamic $\mathbb{P}$.

Assumption 1 (Robust partial coverage data).

We assume that the following robust partial coverage coefficient is finite:

$$C^{\star}_{\mathbb{P}^\star,\mathbf{\Phi}} := \max_{1\le h\le H} \max_{\substack{\mathbb{P}_h\in\mathbf{\Phi}(\mathbb{P}^\star_h)\\ 1\le h\le H}} \mathbb{E}_{(s,a)\sim d^{\pi^{\mathrm{b}}}_{\mathbb{P}^\star,h}}\left[\left(\frac{d^{\pi^\star}_{\mathbb{P},h}(s,a)}{d^{\pi^{\mathrm{b}}}_{\mathbb{P}^\star,h}(s,a)}\right)^2\right] < +\infty, \tag{1}$$

- This only requires that the offline data $d^{\pi^{\mathrm{b}}}_{\mathbb{P}^\star}$ can cover the trajectories induced by the optimal robust policy $d^{\pi^\star}_{\mathbb{P}}$ (for each $\mathbb{P}\in\mathbf{\Phi}(\mathbb{P}^\star)$).
- Weaker and more practical than offline data from generative model or uniformly lower bounded distribution over (s,a).

Main Result: Suboptimality of P^{2}MPO

Theorem 1 (Suboptimality of P^{2}MPO).

Under Assumptions 1 and certain rectangular assumption on RMDP, if P^{2}MPO implements the sub-algorithm `ModelEst`$(\mathcal{D}, \mathcal{P})$ with accuracy $\mathrm{Err}^{\boldsymbol{\Phi}}(N, \delta)$, then with probability at least $1 - 2\delta$,

$$\mathrm{SubOpt}(\widehat{\pi}; s_1) \leq \sqrt{C^{\star}_{\mathbb{P}^\star, \boldsymbol{\Phi}}} \cdot \sum_{h=1}^{H} \sqrt{\mathrm{Err}_h^{\boldsymbol{\Phi}}(N, \delta)}.$$

- $\mathrm{Err}_h^{\boldsymbol{\Phi}}(N, \delta)$ typically achieves a rate of $\widetilde{\mathcal{O}}(N^{-1})$ (see our paper for sub-agorithm design) $\Rightarrow$ P^{2}MPO enjoys $\widetilde{\mathcal{O}}(N^{-1/2})$-suboptimality which is optimal in the number of samples N.
- In tabular setups, the dependence on $C^{\star}_{\mathbb{P}^\star, \boldsymbol{\Phi}}$ is proven inevitable [Shi et al., 2022].

Main Result: Suboptimality of P^{2}MPO

Theorem 1 (Suboptimality of P^{2}MPO).

Under Assumptions 1 and certain rectangular assumption on RMDP, if P^{2}MPO implements the sub-algorithm $\texttt{ModelEst}(\mathcal{D}, \mathcal{P})$ with accuracy $\mathrm{Err}^{\mathbf{\Phi}}(N, \delta)$, then with probability at least $1 - 2\delta$,

$$\mathrm{SubOpt}(\widehat{\pi}; s_1) \leq \sqrt{C^{\star}_{\mathbb{P}^{\star}, \mathbf{\Phi}}} \cdot \sum_{h=1}^{H} \sqrt{\mathrm{Err}_h^{\mathbf{\Phi}}(N, \delta)}.$$

- $\mathrm{Err}_h^{\mathbf{\Phi}}(N, \delta)$ typically achieves a rate of $\widetilde{\mathcal{O}}(N^{-1})$ (see our paper for sub-agorithm design) $\Rightarrow$ P^{2}MPO enjoys $\widetilde{\mathcal{O}}(N^{-1/2})$-suboptimality which is optimal in the number of samples N.
- In tabular setups, the dependence on $C^{\star}_{\mathbb{P}^{\star}, \mathbf{\Phi}}$ is proven inevitable [Shi et al., 2022].

Main Result: Suboptimality of P^{2}MPO

Theorem 1 (Suboptimality of P^{2}MPO).

Under Assumptions 1 and certain rectangular assumption on RMDP, if P^{2}MPO implements the sub-algorithm `ModelEst`$(\mathcal{D}, \mathcal{P})$ with accuracy $\mathrm{Err}^{\mathbf{\Phi}}(N, \delta)$, then with probability at least $1 - 2\delta$,

$$\mathrm{SubOpt}(\widehat{\pi}; s_1) \leq \sqrt{C^{\star}_{\mathbb{P}^{\star}, \mathbf{\Phi}}} \cdot \sum_{h=1}^{H} \sqrt{\mathrm{Err}^{\mathbf{\Phi}}_h(N, \delta)}.$$

- $\mathrm{Err}^{\mathbf{\Phi}}_h(N, \delta)$ typically achieves a rate of $\widetilde{\mathcal{O}}(N^{-1})$ (see our paper for sub-agorithm design) $\Rightarrow$ P^{2}MPO enjoys $\widetilde{\mathcal{O}}(N^{-1/2})$-suboptimality which is optimal in the number of samples N.
- In tabular setups, the dependence on $C^{\star}_{\mathbb{P}^{\star}, \mathbf{\Phi}}$ is proven inevitable [Shi et al., 2022].

Our theory applies to most of known tractable RMDPs for robust offline RL and **new** models by:

- implementing the model estimation subroutine $\texttt{ModelEst}(\mathcal{D}, \mathcal{P})$;
- specifying the robust model estimation accuracy $\mathrm{Err}_h^{\boldsymbol{\Phi}}(N, \delta)$.
- only require "robust partial coverage data"

	Zhou et al. [2021]	Shi and Chi [2022]	Ma et al. [2022]	This Work
$\mathcal{S} \times \mathcal{A}$-rectangular tabular RMDP	✓!	✓	✗	✓
d-rectangular linear RMDP	✗	✗	✓	✓
$\mathcal{S} \times \mathcal{A}$-rectangular factored RMDP	✗	✗	✗	✓
$\mathcal{S} \times \mathcal{A}$-rectangular kernel RMDP	✗	✗	✗	✓
$\mathcal{S} \times \mathcal{A}$-rectangular neural RMDP	✗	✗	✗	✓
$\mathcal{S} \times \mathcal{A}$-rectangular general RMG	NA	NA	NA	✓

Table: ✓: can tackle this model with robust partial coverage data, ✓!: requires full coverage data to solve the model, ✗: cannot tackle the model.

The yellow line denotes the models that are first proposed or proved tractable in this work.

Our theory applies to most of known tractable RMDPs for robust offline RL and **new** models by:

- implementing the model estimation subroutine $\texttt{ModelEst}(\mathcal{D}, \mathcal{P})$;
- specifying the robust model estimation accuracy $\mathrm{Err}_h^{\mathbf{\Phi}}(N, \delta)$.
- only require "robust partial coverage data"

	Zhou et al. [2021]	Shi and Chi [2022]	Ma et al. [2022]	This Work
$\mathcal{S} \times \mathcal{A}$-rectangular tabular RMDP	✓!	✓	✗	✓
d-rectangular linear RMDP	✗	✗	✓	✓
$\mathcal{S} \times \mathcal{A}$-rectangular factored RMDP	✗	✗	✗	✓
$\mathcal{S} \times \mathcal{A}$-rectangular kernel RMDP	✗	✗	✗	✓
$\mathcal{S} \times \mathcal{A}$-rectangular neural RMDP	✗	✗	✗	✓
$\mathcal{S} \times \mathcal{A}$-rectangular general RMG	NA	NA	NA	✓

Table: ✓: can tackle this model with robust partial coverage data, ✓!: requires full coverage data to solve the model, ✗: cannot tackle the model.

The yellow line denotes the models that are first proposed or proved tractable in this work.

Thank You!

Blanchet, J., Lu, M., Zhang, T., & Zhong, H. (2023). Double pessimism is provably efficient for distributionally robust offline reinforcement learning: Generic algorithm and robust partial coverage.

https://arxiv.org/abs/2305.09659

References I

Y. Jin, Z. Yang, and Z. Wang. Is pessimism provably efficient for offline rl? In International Conference on Machine Learning, pages 5084–5096. PMLR, 2021.

X. Ma, Z. Liang, L. Xia, J. Zhang, J. Blanchet, M. Liu, Q. Zhao, and Z. Zhou. Distributionally robust offline reinforcement learning with linear function approximation. arXiv preprint arXiv:2209.06620, 2022.

L. Shi and Y. Chi. Distributionally robust model-based offline reinforcement learning with near-optimal sample complexity. arXiv preprint arXiv:2208.05767, 2022.

L. Shi, G. Li, Y. Wei, Y. Chen, and Y. Chi. Pessimistic q-learning for offline reinforcement learning: Towards optimal sample complexity. arXiv preprint arXiv:2202.13890, 2022.

M. Uehara and W. Sun. Pessimistic model-based offline reinforcement learning under partial coverage. arXiv preprint arXiv:2107.06226, 2021.

Z. Zhou, Z. Zhou, Q. Bai, L. Qiu, J. Blanchet, and P. Glynn. Finite-sample regret bound for distributionally robust offline tabular reinforcement learning. In International Conference on Artificial Intelligence and Statistics, pages 3331–3339. PMLR, 2021.