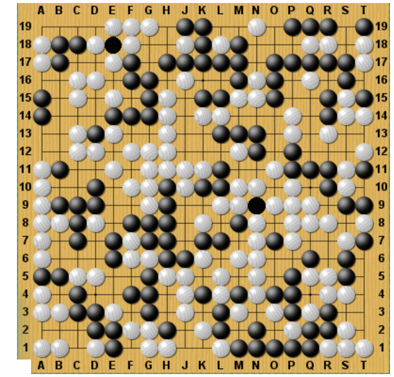
MOPO: Model-based Offline Policy Optimization

Tengyu Ma

Stanford University

Joint work with Tianhe Yu, Garrett Thomas, Lantao Yu, Stefano Ermon, James Zou, Sergey Levine, Chelsea Finn

Sample-Efficiency Challenge in RL



Trials and errors:

- ~~Try the current strategy and collet feedbacks~~
- Use the feedbacks to improve the strategy

millions of games

How to reduce the amount of trials (samples)?

- Model-based RL
- Offline RL, imitation learning
- Meta, multi-task, lifelong, continual RL
- Hierarchical RL
- …

Offline (Batch) Reinforcement Learning

- Given: $\mathcal{B}$ = a collection of trajectories sampled from some policy π^b (under the true dynamics $T^\star$)

$$s_0 \sim D_{s_0} \xrightarrow{\pi^b(s_0)} s_1 \xrightarrow{\pi^b(s_1)} s_2 \xrightarrow{\pi^b(s_2)} s_3 \xrightarrow{\pi^b(s_3)} s_4 \cdots\cdots$$

👍 👎 👎 👍

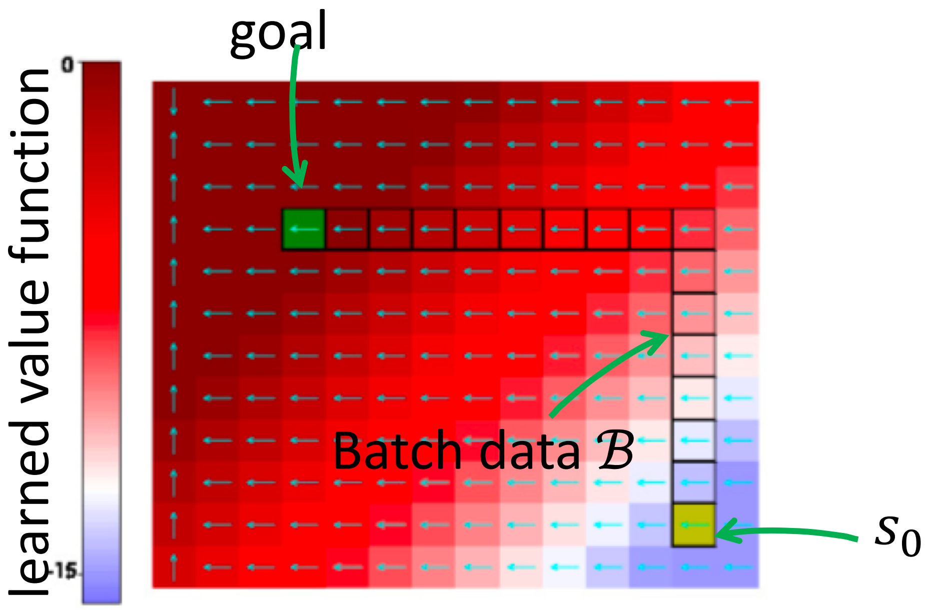
- Reward $r(s_t, a_t) \in \mathbb{R}$ (assumed to be known wlog)
- Goal: learn a policy π that maximizes the expected return

$$\eta^\star(\pi) := \mathbb{E}_{s_0 \sim D_{s_0}}[r(s_0, a_0) + r(s_1, a_1) + r(s_2, a_2) + \cdots]$$

- Offline: interactions with the real environment are not allowed!

The Distribution Shift Issue

- Learning with the batch $\mathcal{B}$ only guarantees accurate predictions on the batch data distribution
- E.g., Q-learning on $\mathcal{B}$ over-estimates the Q-function outside the support of the batch



- Reward = -1 if not reaching the goal
  - $V^{\star} = -$ distance to goal
- Learned value function
  - Correct on $\mathcal{B}$
  - Wrong outside $\mathcal{B}$

Figure from [Learning Self-Correctable Policies and Value Functions from Demonstrations with Negative Sampling . Luo-Xu-M.'19]

A Common Idea: Strong Pessimism/Conservatism

- Stay inside the support of the batch data distribution
  - only visit those (s, a) that you are certain about

A partial list of prior or concurrent work

- BCQ [Fujimoto et al.'19]
- BEAR [Kumar et al.'19]
- BRAC [Wu et al.'19]
- VINS [Luo et al.'19]
- CQL [Kumar et al.'20]
- ...

Q: Can we risk leaving the support of the batch data in exchange for higher return?

Simplification: Offline Multi-Arm Bandit

- Can only pull your arm once!

Restaurant 1	Restaurant 2	Restaurant 3	Restaurant 4
1 reviews 4.7 stars	10 reviews 4.65 stars	100 reviews 4.4 stars	10K reviews 4.3 stars

- “Strong conservatism”: only considering restaurants with prob. > 2% in the batch data
  - Choice = Restaurant 4

Milder Conservatism: Trading off Return with Risk

Restaurant 1	Restaurant 2	Restaurant 3	Restaurant 4
1 reviews	10 reviews	100 reviews	10K reviews
4.7 stars	4.65 stars	4.4 stars	4.3 stars

Confidence interval (error bar $\approx 1/\sqrt{n}$):

[3.7, 5.7]	[4.33, 4.94]	[4.3, 4.5]	[4.29, 4.31]

$$\max_{a} \text{lower-confidence}(a)$$

Back to Offline Reinforcement Learning

Step 1: build uncertainty quantification of return

$$\eta^\star(\pi) \in [\hat{\eta}(\pi) \pm e(\pi)]$$

Step 2: maximize the lower confidence bound

$$\max_\pi \hat{\eta}(\pi) - e(\pi)$$

Step 1: Uncertainty Quantification (UQ) For the Return

- A model-based approach
  - UQ for the learned dynamics → UQ for the return

- Learn a dynamical model $\widehat{T}$ on the batch data which is assumed to deterministic (for now)

- Calibrated model: assume error estimator $u(\cdot,\cdot)$ for $\widehat{T}$ satisfying
$$||\widehat{T}(s,a) - T^{\star}(s,a)|| \leq u(s,a)$$
- Assume the value function $V^{\pi,T^{\star}}$ is c-Lipschitz

Theorem: Let $\hat{\eta}(\pi)$ be the return on the learned dynamics, then
$$\eta^{\star}(\pi) \in [\hat{\eta}(\pi) \pm e(\pi)]$$
$$\text{where } \mathrm{e}(\pi) = \frac{c\gamma}{1-\gamma} \cdot \mathbb{E}_{(s,a)\sim\pi,\widehat{T}}[u(s,a)]$$

Unified Approach for Stochastic Dynamics

- Assume $V^{\pi,T^\star} \in c \cdot \mathcal{F}$ where $c \in \mathbb{R}$
- Assume error estimator $u(\cdot,\cdot)$ for learned (stochastic) dynamics T sat.

$$d_{\mathcal{F}}\big(T(s,a), T^\star(s,a)\big) \leq u(s,a)$$

where $d_{\mathcal{F}}$ is integral probability metric (IPM) between two dist. w.r.t $\mathcal{F}$.

$$d_{\mathcal{F}}(P,Q) := \sup_{f\in\mathcal{F}} \left| \mathop{\mathbb{E}}_{X\sim P}[f(X)] - \mathop{\mathbb{E}}_{Y\sim Q}[f(Y)] \right|$$

- If $V^{\pi,T^\star}$ is L-Lipschitz, then $d_{\mathcal{F}}$ = the Wasserstein distance (and ℓ_2-distance if dynamics is deterministic)
- If $V^{\pi,T^\star}$ is bounded, then $d_{\mathcal{F}}$ = TV-distance.
- If $V^{\pi,T^\star}$ is in some kernel space, then $d_{\mathcal{F}}$ = maximum mean discrepancy (MMD).

Lemma: under the assumption above, we have

$$\eta^\star(\pi) \in [\hat{\eta}(\pi) \pm e(\pi)]$$

for $e(\pi) = \mathbb{E}_{(s,a)\sim\pi,T}[\lambda \cdot u(s,a)]$ with $\lambda = \frac{c\gamma}{1-\gamma}$.

Proof Sketch

- $\eta^\star(\pi) - \hat{\eta}(\pi)$

telescoping sum

$$= \gamma \mathbb{E}_{(s,a)\sim\pi,\hat{T}}\Big[\mathbb{E}_{s'\sim\hat{T}(s,a)}[V^{\pi,T^\star}(s')] - \mathbb{E}_{s'\sim T^\star(s,a)}[V^{\pi,T^\star}(s')]\Big]$$

Def. of IPM

Lemma: under the assumption above, we have

$$\eta^\star(\pi) \in [\hat{\eta}(\pi) \pm e(\pi)]$$

for $e(\pi) = \mathbb{E}_{(s,a)\sim\pi,T}[\lambda \cdot u(s,a)]$ with $\lambda = \frac{c\gamma}{1-\gamma}$.

MOPO: Model-based Policy Opt. with Reward Penalty

Step 2: Optimize $\hat{\eta}(\pi) - e(\pi) = \mathbb{E}_{(s,a)\sim\pi,\hat{T}}[r(s,a) - \lambda \cdot u(s,a)]$

A. Define a MDP $\widetilde{M}$ with the learned dynamics $\hat{T}$ and penalized reward

$$\tilde{r}(s,a) = r(s,a) - \lambda \cdot u(s,a)$$

B. Find the optimal policy of $\widetilde{M}$ with off-the-shelf RL algo.

- Implementation of UQ: use ensemble as a heuristic for $u(s,a)$

Characterizing the Tradeoff between the Gain and Risk of Leaving Batch Data Support

Theorem: Let $\epsilon(\pi) = \mathbb{E}_{(s,a)\sim\pi,\hat{T}}[u(s,a)]$ which captures the risk. The policy $\hat{\pi}$ found by MOPO satisfies:

$$\eta^\star(\hat{\pi}) \geq \sup_{\pi}\{\eta^\star(\pi) - 2\lambda \cdot \epsilon(\pi)\}$$

Two ends of the spectrum:

- Taking $\pi = \pi^b$, then $\eta^\star(\hat{\pi}) \geq \eta^\star(\pi^b) - \overbrace{2\lambda\epsilon(\pi^b)}^{\approx 0 \text{ bc. no dist. shift}} \approx \eta^\star(\pi^b)$

- Taking $\pi = \pi^\star$, then $\eta^\star(\hat{\pi}) \geq \eta^\star(\pi^\star) - \underbrace{2\lambda\epsilon(\pi^\star)}_{\text{depends on how far } \pi^\star \text{ is from the batch data dist.}}$

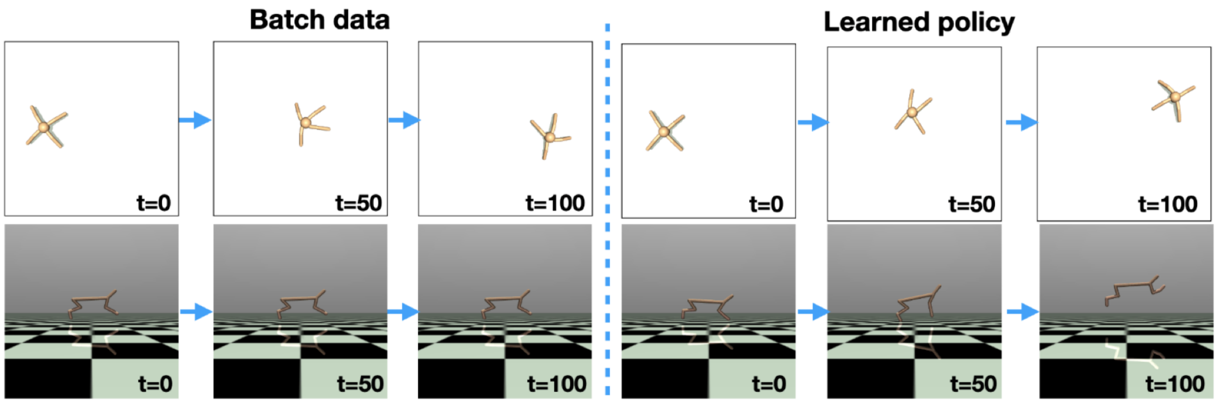
Evaluation on D4RL dataset

➢ [Fu et al.20'] D4RL: Datasets for deep data-driven reinforcement learning

Dataset type	Environment	Batch Mean	Batch Max	MOPO (ours)	MBPO	SAC	BEAR	BRAC-v
random	halfcheetah	-303.2	-0.1	**3679.8** $\pm$ 70.7	3533.0 $\pm$ 201.8	3502.0	2885.6	3207.3
random	hopper	299.26	365.9	**412.8** $\pm$ 30.7	126.6 $\pm$ 173.9	347.7	289.5	370.5
random	walker2d	0.9	57.3	**596.3** $\pm$ 121.8	395.9 $\pm$ 371.7	192.0	307.6	23.9
medium	halfcheetah	3953.0	4410.7	4706.9 $\pm$ 61.1	3230.0 $\pm$ 2543.6	-808.6	4508.7	**5365.3**
medium	hopper	1021.7	3254.3	840.9 $\pm$ 99.3	137.8 $\pm$ 87.5	5.7	**1527.9**	1030.0
medium	walker2d	498.4	3752.7	645.5 $\pm$ 464.8	582.6 $\pm$ 348.8	44.2	1526.7	**3734.3**
mixed	halfcheetah	2300.6	4834.2	**6418.3** $\pm$ 47.4	5598.4 $\pm$ 1285.1	-581.3	4211.3	5413.8
mixed	hopper	470.5	1377.9	**2988.7** $\pm$ 186.3	1599.2 $\pm$ 969.6	93.3	802.7	5.3
mixed	walker2d	358.4	1956.5	**1963.5** $\pm$ 383.8	1021.8 $\pm$ 585.8	87.8	495.3	44.5
med-expert	halfcheetah	8074.9	12940.2	**6913.5** $\pm$ 2793.0	929.6 $\pm$ 903.2	-55.7	6132.5	5342.4
med-expert	hopper	1850.5	3760.5	1663.5 $\pm$ 1375.6	**1803.6** $\pm$ 1102.4	32.9	109.8	5.1
med-expert	walker2d	1062.3	5408.6	2527.1 $\pm$ 879.8	351.7 $\pm$ 170.6	-5.1	1193.6	**3058.0**

Out-of-distribution Offline RL Tasks

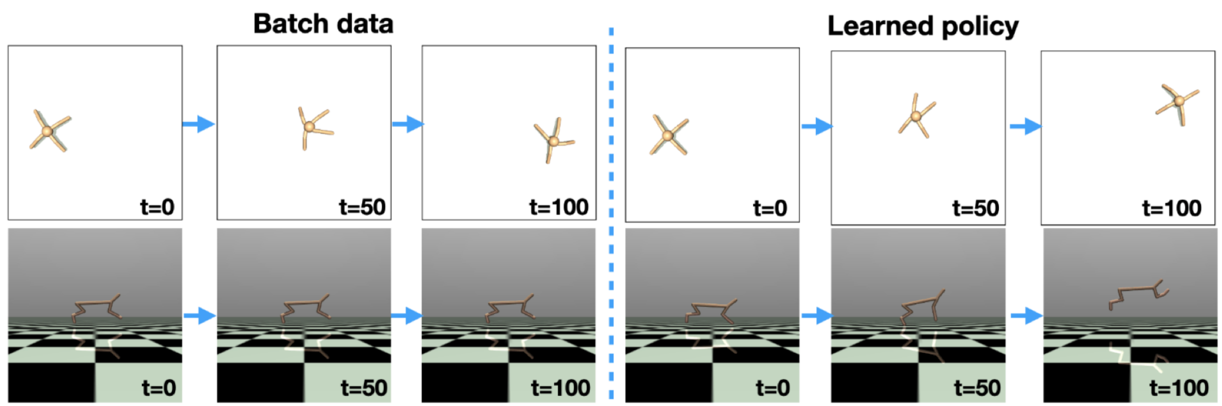
- Situations where the agent has to take the risk of leaving the support of the batch data to achieve high reward



- ant-angle
  - batch: ant runs forward
  - Task: ant is supposed to run to the direction with degree 30
- cheetah-jump:
  - batch: cheetah runs forward
  - Task: cheetah is supposed to jump

Out-of-distribution Offline RL Tasks

- Situations where the agent has to take the risk of leaving the support of the batch data to achieve high reward



Environment	Batch Mean	Batch Max	MOPO (ours)	MBPO	SAC	BEAR	BRAC-p	BRAC-v
halfcheetah-jump	-1022.6	1808.6	**4016.6±144**	2971.4±1262	-3588.2±1436	16.8±60	1069.9±232	871±41
ant-angle	866.7	2311.9	**2256.0±288**	13.6±66	-966.4±778	1658.2±16	1806.7±265	2333±139

Summary

This talk:

- MOPO: offline model-based RL with a reward penalty from uncertainty quantification

Open questions:

- Tighter uncertainty quantification?
- Less conservative than optimizing lower confidence bound?

Ads of RL work by my group:

- Model-based vs model-free through the lens of expressivity:
  - On the Expressivity of Neural Networks for Deep Reinforcement Learning. ICLM 2020
- Addressing distribution shift in meta-RL:
  - Model-based Adversarial Meta-Reinforcement Learning. to appear at NeuRIP'20