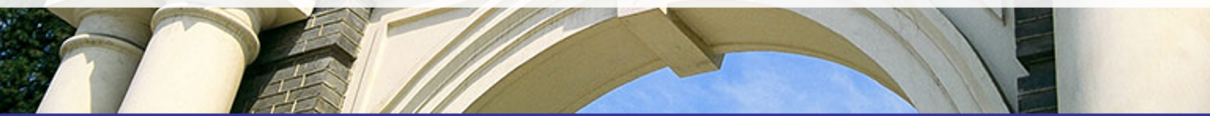
Bayesian Design Principles for Offline-to-Online Reinforcement Learning

Hao Hu*, Yiqin Yang*, Jianing Ye, Ziqing Mai, Yujing Hu, Tangjie Lv, Changjie Fan, Qianchuan Zhao, Chongjie Zhang

August 13, 2024



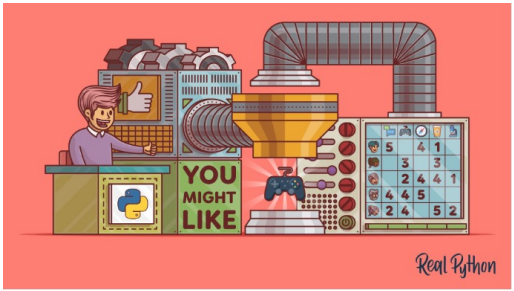
Machine Intelligence Group

Tsinghua University Institute for Interdisciplinary Information Sciences

Offline Reinforcement Learning

Offline RL (He and Hou, 2020; Kumar et al., 2020) has been popular in recent years, since it can

- Reduce the exploration constant
- Facilitate data reuse



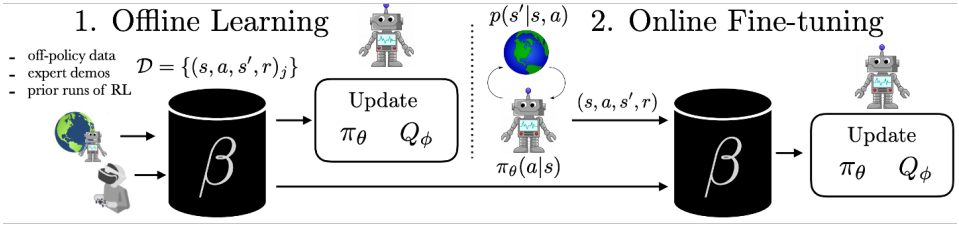
Offline-to-Online RL

However...

- Offline agents are still suboptimal
- Offline agents generalize poorly

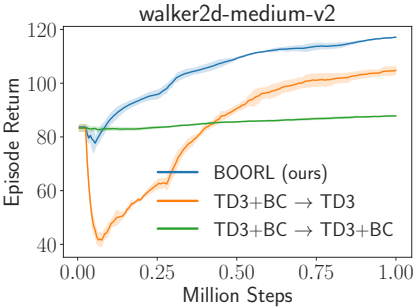
Offline-to-Online RL

A natural paradigm:



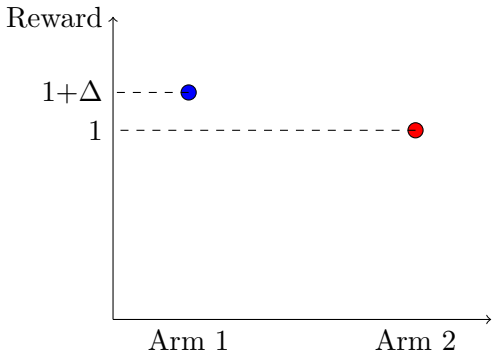
The exploration-exploitation dilemma in off-to-on RL

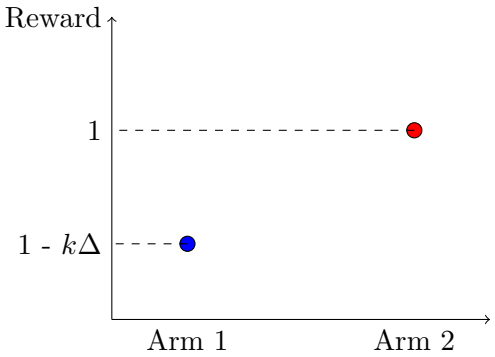
- offline algorithms learn slowly due to a lack of exploration (green).
- online algorithms suffer from a sudden drop due to radical exploration (orange).



The Unsolvable Dilemma

Consider the two following bandits:





A Bayesian Point of View

Preliminaries

Let $\mathcal{H}_{k,h} = (s_{1,1}, a_{1,1}, r_{1,1}, \ldots, s_{k,h-1}, a_{k,h-1}, r_{k,h-1}, s_{k,h})$ be all the history up to step h of episode k. We use subscript k, h to indicate quantities conditioned on $\mathcal{H}_{k,h}$, i.e. $\mathbb{P}_{k,h} = \mathbb{P}(\cdot|\mathcal{H}_{k,h}), \mathbb{E}_{k,h}[\cdot] = \mathbb{E}[\cdot|\mathcal{H}_{k,h}]$. The filtered mutual information is defined as

$$I_{k,h}(X;Y) = D_{\mathrm{KL}}(\mathbb{P}_{k,h}(X,Y)||\mathbb{P}_{k,h}(X)\mathbb{P}_{k,h}(Y)),$$

which is a random variable of $\mathcal{H}_{k,h}$. For a horizon dependent quantity $f_{k,h}$, we define $\mathbb{E}_k[f_k] = \sum_{h=1}^{H} \mathbb{E}_{k,h}[f_{k,h}]$ and similarly for $\mathbb{P}_k$.

The information ratio (Russo and Van Roy, 2016) as the ratio between the expected single step regret and the expected reduction in entropy of the unknown parameter as follows

Definition (Information Ratio (Russo and Van Roy, 2016))

The information ratio $\Gamma_{k,h}$ given history $\mathcal{H}_{k,h}$ is the minimum value Γ such that the following event

$$|Q_{w_h}(s,a) - \mathbb{E}Q_{w_h}(s,a)| \leq \frac{\Gamma}{2}\sqrt{I_{k,h}(w_h; r_h, s_{h+1} \,|\, s, a)}, \tag{1}$$

holds for all $h \in [H], s \in \mathcal{S}, a \in \mathcal{A}$ with probability $1 - \delta/2$.

A Bayesian Point of View

Theorem

Then the per-episode regret of Thompson Sampling and UCB agents satisfies

$$\mathbb{E}_k[\Delta_k] \leq \sum_{h=1}^{H} \Gamma_{k,h} \sqrt{I_{k,h}(w_h; a_{k,h}, r_{k,h}, s_{k,h+1})} + 2\delta H^2, \tag{2}$$

where $a_{k,h} \sim \pi_{k,h}$. *Similarly, the per-episode regret of Thompson Sampling and LCB agents satisfies*

$$\mathbb{E}_k[\Delta_k] \leq \sum_{h=1}^{H} \Gamma_{k,h} \sqrt{I_{k,h}(w_h; a_h^*, r_{k,h}, s_{k,h+1})} + 2\delta H^2, \tag{3}$$

where $a_h^* \sim \pi_h^*$.

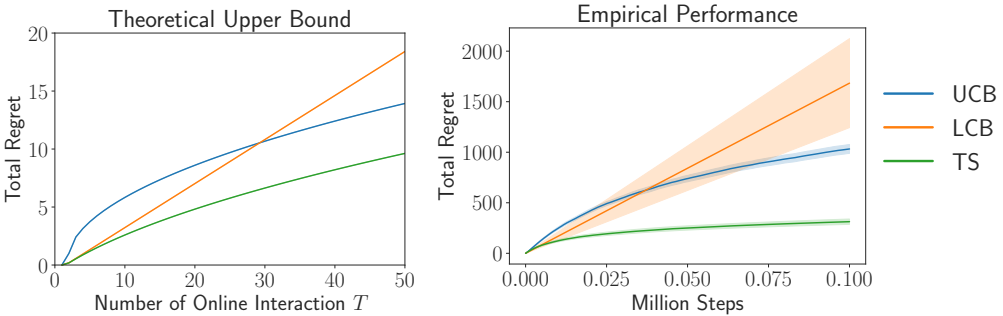


Figure 1: Theoretical upper bound in Theorem 0.2 and experiment results on Bernoulli bandits. The performance of a Bayesian approach matches the performance of LCB at an early stage by using prior knowledge in the dataset properly and matches the performance of UCB in the run by allowing efficient exploration. Therefore, a realistic Bayesian agent performs better than both optimistic UCB and pessimistic LCB agents.

Setting	Doctrine	Algorithm
Online Learning	Optimism	TS, UCB
Offline Learning	Pessimism	TS, LCB
Offline-to-online	Realism	TS

Table 1: A taxonomy of the doctrines in different settings of reinforcement learning. a Bayesian approach like TS is generally suitable for online, offline and offline-to-online settings, and is the only one that works in the offline-to-online setting.

Algorithm Design

Algorithm 1 BOORL, Offline Phase

Require: Ensemble size N, offline dataset $\mathcal{D}^{\text{off}}$, masking distribution M
Initialize parameters of N independent TD3+BC agents $\{Q_{\theta_i}, \pi_{\phi_i}\}_{i=1}^{N}$
for $i = 1, \cdots, N$ **do**
  Sample bootstrap mask $m \sim M$ (e.g., Bernoulli distribution)
  Add m to $\mathcal{D}^{\text{off}}$ as $\mathcal{D}_i^{\text{off}}$
  for each training iteration **do**
    Sample a random minibatch $\{\tau_j\}_{j=1}^{B} \sim \mathcal{D}_i^{\text{off}}$
    Calculate $L_{\text{critic}}^{\text{offline}}(\theta_i)$ and update θ_i
    Calculate $L_{\text{actor}}^{\text{offline}}(\phi_i)$ and update ϕ_i
  end for
end for
Return $\{Q_{\theta_i}, \pi_{\phi_i}\}_{i=1}^{N}$

Algorithm Design

Algorithm 2 BOORL, Online Phase

Require: $\{Q_{\theta_i}, \pi_{\phi_i}\}_{i=1}^{N}$, offline dataset $\mathcal{D}^{\text{off}}$, empty online replay buffer $\mathcal{D}^{\text{on}}$
for each iteration **do**
  for step $t = 1, \cdots, T$ **do**
    Construct distribution $p_i = \frac{\exp(Q_{\theta_i}(s_t, \pi_{\phi_i}(s_t)))}{\sum_j \exp(Q_{\theta_j}(s_t, \pi_{\phi_j}(s_t)))}$
    Pick an policy to act $a_t \sim \pi_{\phi_n}(\cdot \mid s_t)$ by sampling index n based on p_i
    Store transition (s_t, a_t, r_t, s_{t+1}) in $\mathcal{D}^{\text{on}}$
    Sample minibatch B from $\mathcal{D}^{\text{off}}$ and $\mathcal{D}^{\text{on}}$
    for $i = 1, \cdots, N$ **do**
      Calculate $L_{\text{critic}}^{\text{online}}(\theta_i), L_{\text{actor}}^{\text{online}}(\phi_i)$ with minibatch B and update θ_i, ϕ_i
    end for
  end for
end for

Experimental Results

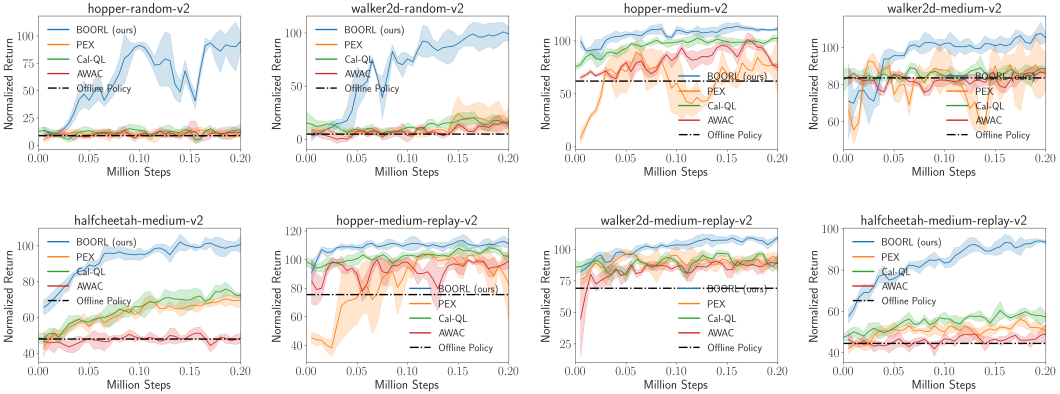


Figure 2: Experiments between several baselines and BOORL within 0.2M time steps. The reference line is the performance of TD3+BC. The experimental results are averaged with five random seeds.

Experimental Results

Task	ODT	PEX	Cal-QL	BOORL	RLPD
hopper-rand	10.1→30.8	7.6→10.1	9.3→11.9	8.8→75.7	**84.1**
hopper-med	66.9→97.5	63.8→78.6	75.8→100.6	61.9→**109.8**	107.3
hopper-exp	108.1→**110.7**	102.4→96.6	94.8→**110.3**	111.5→**109.2**	100.4
cheetah-rand	1.1→2.2	9.6→61.2	22.0→45.1	10.7→**97.7**	63.0
cheetah-med	42.7→42.1	47.3→67.8	48.0→72.3	47.9→**98.7**	90.5
cheetah-exp	87.3→94.3	90.5→95.5	64.5→92.1	97.5→**98.4**	93.2
antmaze-u	56.6→83.5	81.6→**100.0**	78.5→**100.0**	81.7→**100.0**	95.6
antmaze-m-p	0.0→0.0	68.6→90.8	59.4→91.9	50.6→**100.0**	96.3
antmaze-l-p	0.0→0.0	49.9→68.2	24.2→55.9	61.0→75.8	**81.6**
δ_{sum} (0.2M)	146.0	326.8	392.0	**698.1**	-

Experimental Results

Task	Type	BOORL	Bayesian	δ	Hybrid RL	δ
Hopper	random	75.7±1.3	85.4±3.3	-9.7	75.2±3.9	0.5
	medium	109.8±1.6	109.6±1.5	0.2	91.4±1.2	18.4
	medium-replay	111.1±0.3	110.6±0.6	0.5	103.5±2.7	7.6
Walker2d	random	93.6±4.4	92.4±4.7	1.2	15.4±0.8	78.2
	medium	107.7±0.5	96.5±3.5	11.2	86.4±0.4	21.3
	medium-replay	114.4±0.9	103.7±2.1	10.7	99.7±2.4	14.7
Halfcheetah	random	97.7±1.1	94.5±4.2	3.2	85.2±0.5	12.5
	medium	98.7±0.3	97.7±0.5	1.0	80.3±0.2	18.4
	medium-replay	91.5±0.9	90.5±0.5	1.0	84.8±1.0	6.7

Table 2: Ablation results on Mujoco tasks with the normalized score metric.

Thanks for Listening!

References I

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In International Conference on Machine Learning, pages 2052–2062. PMLR, 2019.

Qiang He and Xinwen Hou. Popo: Pessimistic offline policy optimization. arXiv preprint arXiv:2012.13682, 2020.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. Advances in Neural Information Processing Systems, 33:1179–1191, 2020.

Daniel Russo and Benjamin Van Roy. An information-theoretic analysis of thompson sampling. The Journal of Machine Learning Research, 17(1):2442–2471, 2016.

Yue Wu, Shuangfei Zhai, Nitish Srivastava, Joshua Susskind, Jian Zhang, Ruslan Salakhutdinov, and Hanlin Goh. Uncertainty weighted actor-critic for offline reinforcement learning. arXiv preprint arXiv:2105.08140, 2021.

Tianhe Yu, Aviral Kumar, Rafael Rafailov, Aravind Rajeswaran, Sergey Levine, and Chelsea Finn. Combo: Conservative offline model-based policy optimization. Advances in Neural Information Processing Systems, 34, 2021.