On the Value of Interaction in Imitation Learning

Presentor: Tian Xu
xut@lamda.nju.edu.cn

Nanjing University, Nanjing, China

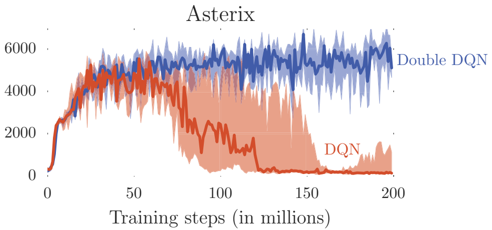
Mainly based on the paper:
Nived, Rajaraman, et al. "On the Value of Interaction and Function Approximation in Imitation Learning." NeurIPS, 2021.

March 15, 2022

What to Expect from This Talk?

- A big picture on imitation learning (IL).
- Understand the fundamental difference between the offline setting and active setting in IL.

Background of Imitation Learning

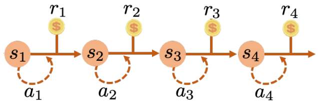


(a) Double DQN requires million interactions to solve Atari games [van Hasselt et al., 2016].

(b) Robot directly learns from human demonstrations.

- Two Challenges when applying RL in real world.
  - It often requires a large amount of environment interactions.
  - It's hard and inefficient to design proper reward function for each particular task.
- In some real-world scenarios, it is easy to obtain expert-level demonstrations.

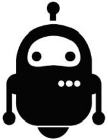
Markov Decision Process



Markov Decision Process

- Consider a finite episodic Markov Decision Process $\left(\mathcal{S}, \mathcal{A}, H, \{P_h\}_{h\in[H]}, \{r_h\}_{h\in[H]}, \rho\right)$.
- A policy π is a collection of functions $\pi_h : \mathcal{S} \to \Delta(\mathcal{A})$ for all $h \in [H]$.
- The value function and Q-value function of π:
  $V_h^\pi(s) \triangleq \mathbb{E}\left[\sum_{h'=h}^{H} r_{h'}(s_{h'}, a_{h'}) \mid s_h = s, \pi\right], Q_h^\pi(s,a) \triangleq \mathbb{E}\left[\sum_{h'=h}^{H} r_{h'}(s_{h'}, a_{h'}) \mid s_h = s, a_h = a, \pi\right].$
- The value of policy π: $V(\pi) = \mathbb{E}_{s_1\sim\rho}[V_1^\pi(s_1)]$.
- The state-action distribution induced by π $P_h^\pi(s,a) = \mathbb{P}(s_h = s, a_h = a|\pi)$.

Imitation Learning (IL)



Learner

$\pi(a|s)$

Expert

$(s, a) \sim \pi_{\mathrm{E}}$

- The expert demonstrations is a set of trajectories $D = \{(s_1^i, a_1^i, s_2^i, \cdots, s_H^i, a_H^i)\}_{i=1}^m$, where actions are the output of expert policy π^{E}, which is assumed to be deterministic.
- Agent directly learns a policy from D without explicit rewards.
- The target in IL: $\min_\pi V(\pi_E) - V(\pi) \quad \Longleftrightarrow \quad \max_\pi V(\pi)$.

Settings

There are mainly three settings in IL.

- **Offline**: Provided with expert dataset, the learner is not allowed to interact with the MDP.
- **Active**: Without expert dataset in advance, the learner is allowed to interact with the MDP for m episodes and query an oracle to the expert actions on states collected by the learner.
- **Known-transition**: With expert dataset in advance, the learner additionally knows the MDP transition function.
  - **A "weaker" version**: the learner can interact with the MDP a finite number of times.

We focus on the offline and active setting.

A Big Picture of IL

There are mainly two classes of IL algorithms: Behavioral Cloning (BC) based and Adversarial Imitation Learning (AIL) based methods.

- BC [Pomerleau, 1991] minimizes the action discrepancy on the expert's state distribution.

$$\min_{\pi} \mathcal{L}_{\mathsf{bc}} \left(\pi, \pi^{\mathsf{E}}\right) := \frac{1}{H} \sum_{t=1}^{H} \mathbb{E}_{s_t \sim P_t^{\pi^{\mathsf{E}}}(\cdot)} \left[\mathbb{E}_{a \sim \pi_t(\cdot|s_t)} \left[\mathbb{I}\left(a \neq \pi_t^{\mathsf{E}}\left(s_t\right)\right)\right]\right],$$

  Remark: BC is applied under the offline setting.

- Another BC-based method DAgger [Ross et al., 2011] minimizes the action discrepancy on the learner's state distribution.

$$\min_{\pi} \mathcal{L}_{\mathsf{dagger}} \left(\pi, \pi^{\mathsf{E}}\right) := \frac{1}{H} \sum_{t=1}^{H} \mathbb{E}_{s_t \sim P_t^{\pi}(\cdot)} \left[\mathbb{E}_{a \sim \pi_t(\cdot|s_t)} \left[\mathbb{I}\left(a \neq \pi_t^{\mathsf{E}}\left(s_t\right)\right)\right]\right],$$

  Remark: DAgger is applied under the active setting.

A Big Picture of IL

- AIL based methods minimize the discrepancy between state-action distributions with some divergence d. $\min_\pi \sum_{h=1}^{H} d(P_h^{\pi}, P_h^{\pi^{\mathsf{E}}})$.
- Optimizing this objective requires the knowledge of transitions and hence AIL is often applied under the known-transition setting or its weaker version.
- GAIL [Ho and Ermon, 2016] is a famous AIL method and minimizes the objective in an adversarial manner like GAN [Goodfellow et al., 2014].

Settings	Remarkable Algorithms
Offline	BC
Active	DAgger, AGGREVATE [Ross and Bagnell, 2014]
Known-transition (weaker version)	GAIL, DAC [Kostrikov et al., 2019]

Theoretical Guarantees

An IL problem is specified by $(\mathcal{M}, \pi^{\mathsf{E}})$. For an IL algorithm *Alg*, we measure its performance on $(\mathcal{M}, \pi^{\mathsf{E}})$ by $V(\pi^{\mathsf{E}}) - \mathbb{E}\left[V(\overline{\pi})\right]$, where $\overline{\pi}$ is the output of *Alg*.

Definition 1: Algorithm-dependent upper bound

Consider *Alg*, for any IL problems $(\mathcal{M}, \pi^{\mathsf{E}})$, $V(\pi^{\mathsf{E}}) - \mathbb{E}\left[V(\overline{\pi})\right] \leq \text{Ploy}(|\mathcal{S}|, H, 1/m)$.

Definition 2: Setting-dependent lower bound

For any *Alg* under some specific setting (e.g., offline), there exists a hard IL problem $(\mathcal{M}, \pi^{\mathsf{E}})$, $V(\pi^{\mathsf{E}}) - \mathbb{E}\left[V(\overline{\pi})\right] \geq \text{Ploy}(|\mathcal{S}|, H, 1/m)$.

- Upper bound measures the performance of an algorithm and lower bound measures the hardness of some specific setting.
- If an algorithm's upper bound matches the lower bound, this algorithm is minimax optimal.

Limitations of the Worst-Case Analysis

Settings	Lower Bound	Upper Bound
Offline	$\Omega\left(\frac{\lvert\mathcal{S}\rvert H^2}{m}\right)$	BC: $\mathcal{O}\left(\frac{\lvert\mathcal{S}\rvert H^2}{m}\right)$
Active	$\Omega\left(\frac{\lvert\mathcal{S}\rvert H^2}{m}\right)$	BC: $\mathcal{O}\left(\frac{\lvert\mathcal{S}\rvert H^2}{m}\right)$

Table: Summary of existing results on the lower bound and upper bound [Rajaraman et al., 2020].

- The H^2 dependence on BC's upper bound is known as the compounding errors issue [Ross and Bagnell, 2010]. The lower bound under the active setting implies that the compounding error issue is unavoidable even when the learner can interact with the MDP.
- From the worst-case analysis (i.e., for all IL problems), we cannot see the benefits from online interactions in the active setting.
- The worst-case analysis cannot help explain that DAgger, which operates under the active setting, often performs better than BC in practice.

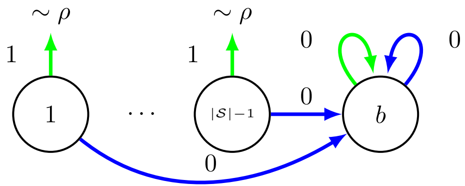
Main Contributions

Settings	Lower Bound	Upper Bound
Offline	$\Omega\left(\frac{\lvert\mathcal{S}\rvert H^2}{m}\right)$	BC: $\mathcal{O}\left(\frac{\lvert\mathcal{S}\rvert H^2}{m}\right)$
Active	$\Omega\left(\frac{\mu\lvert\mathcal{S}\rvert H}{m}\right)$	$\mathcal{O}\left(\frac{\mu\lvert\mathcal{S}\rvert H}{m}\right)$

Table: Summary of results on the lower bound and upper bound under the μ-recoverability assumption.

- The authors study a class of IL problems under the μ-recoverability assumption.
- They develop an algorithm with an upper bound $\mathcal{O}\left(\frac{\mu\lvert\mathcal{S}\rvert H}{m}\right)$ under the active setting, which provably mitigates the compounding errors issue.
- They establish lower bounds $\Omega\left(\frac{\lvert\mathcal{S}\rvert H^2}{m}\right)$ and $\Omega\left(\frac{\mu\lvert\mathcal{S}\rvert H}{m}\right)$ for offline and active setting, resp.
- This result shows the benefits from online interactions and establishes a clear separation between offline and active setting.

Revisit the Hard Instance in the Worst-case Analysis



The first $|\mathcal{S}| - 1$ states are good states and the last state is a bad absorbing state. Green arrows and blue arrows indicate the transitions via expert actions and non-expert actions. ρ is a state distribution which supports on good states. The digits besides arrows indicates rewards.

- This hard instance is strict in the sense that if the expert policy visits the bad state accidentally, it is never able to "recover" and return to good states.
- In practical situations (e.g., driving a car), experts often can recover and collect a high reward even if a mistake is made locally.

μ-recoverability Assumption

Definition 3: μ-recoverability

An IL problem is said to satisfy μ-recoverability if for each $t \in [H]$ and $s \in \mathcal{S}$,

$$Q_t^{\pi^{\mathsf{E}}}(s, \pi_t^{\mathsf{E}}(s)) - Q_t^{\pi^{\mathsf{E}}}(s, a) \leq \mu, \forall a \in \mathcal{A}.$$

- If π^{E} plays an non-expert action a at any state s in timestep t and returns to choosing the expert action afterwards, the expected reward collected is less by at most μ.
- Note that $\mu \leq H$. Sanity check: when $\mu = H$, IL problems with μ-recoverability assumption reduce to all IL problems; the results under μ-recoverability assumption reduce to the worst-case results.
- In the last hard instance, $\mu = H$.

Upper Bound Under the Active Setting

Theorem 1

Consider the active setting, under the μ-recoverability condition, we can construct an algorithm which outputs $\overline{\pi}$ and have

$$V(\pi^{\mathsf{E}}) - \mathbb{E}\left[V(\overline{\pi})\right] \precsim \frac{\mu|\mathcal{S}|H}{m}.$$

- The policy value gap has a linear dependence on H, which provably breaks the compounding errors barrier in BC.

Upper Bound Analysis

Offline, BC: $\mathcal{O}\left(\frac{|\mathcal{S}|H^2}{m}\right)$ V.S. Active: $\mathcal{O}\left(\frac{\mu|\mathcal{S}|H}{m}\right)$

Proposition 1: Reduction Framework [Ross et al., 2011]

Consider IL problems with μ-recoverability assumption, for any π,

$$V(\pi^{\mathsf{E}}) - V(\pi) \leq H \sum_{t=1}^{H} \mathbb{E}_{s_t \sim P_t^{\pi^{\mathsf{E}}}(\cdot)} \left[\mathbb{E}_{a \sim \pi_t(\cdot|s_t)} \left[\mathbb{I}\left(a \neq \pi_t^{\mathsf{E}}(s_t)\right)\right]\right]$$

$$V(\pi^{\mathsf{E}}) - V(\pi) \leq \mu \underbrace{\sum_{t=1}^{H} \mathbb{E}_{s_t \sim P_t^{\pi}(\cdot)} \left[\mathbb{E}_{a \sim \pi_t(\cdot|s_t)} \left[\mathbb{I}\left(a \neq \pi_t^{\mathsf{E}}(s_t)\right)\right]\right]}_{L(\pi, P^{\pi}, \pi^{\mathsf{E}})}$$

- [Ross et al., 2011] does not show how small is $L(\pi, P^{\pi}, \pi^{\mathsf{E}})$. This work designs an algorithm and shows that $L(\pi, P^{\pi}\pi^{\mathsf{E}})$ can be minimized up to $\mathcal{O}\left(\frac{|\mathcal{S}|H}{m}\right)$.

Algorithm Design

Target: find a policy $\overline{\pi}$ s.t. $\mathbb{E}\left[L(\overline{\pi}, P^{\overline{\pi}}, \pi^{\mathsf{E}})\right] \leq \mathcal{O}\left(\frac{|\mathcal{S}|H}{m}\right)$.

$L(\pi, P^{\pi}, \pi^{\mathsf{E}}) = \sum_{t=1}^{H} \mathbb{E}_{s_t \sim P_t^{\pi}(\cdot)}\left[\mathbb{E}_{a \sim \pi_t(\cdot|s_t)}\left[\mathbb{I}\left(a \neq \pi_t^{\mathsf{E}}(s_t)\right)\right]\right]$

- Online learning: find a sequence of policies $\overline{\pi}^1, \cdots, \overline{\pi}^m$ and output the mixture policy $\overline{\pi}$.
- The mixture policy satisfies that $L(\overline{\pi}, P^{\overline{\pi}}, \pi^{\mathsf{E}}) = \frac{1}{m}\sum_{i=1}^{m} L(\overline{\pi}^i, P^{\overline{\pi}^i}, \pi^{\mathsf{E}})$.
- Regard $L(\pi, P^{\overline{\pi}^i}, \pi^{\mathsf{E}})$ as an objective of π and $\min_{\pi} \sum_{i=1}^{m} L(\pi, P^{\overline{\pi}^i}, \pi^{\mathsf{E}}) = \sum_{i=1}^{m} L(\pi^{\mathsf{E}}, P^{\overline{\pi}^i}, \pi^{\mathsf{E}}) = 0$.
- Now the target is changed to upper bound this online learning regret:

$$\sum_{i=1}^{m} L(\overline{\pi}^i, P^{\overline{\pi}^i}, \pi^{\mathsf{E}}) - \min_{\pi} \sum_{i=1}^{m} L(\pi, P^{\overline{\pi}^i}, \pi^{\mathsf{E}}) \leq \mathcal{O}\left(|\mathcal{S}|H\right).$$

Online Learning Framework

Algorithm 1 Online Learning Framework

1: **Input:** Uniformly initialized policy $\overline{\pi}^1$
2: **for** $i = 1, 2, \cdots, m$ **do**
3: The learner takes policy $\overline{\pi}^i$ and observes objective function $L^i(\pi) = L(\pi, P^{\overline{\pi}^i}, \pi^{\mathsf{E}})$
4: The learner updates the policy $\overline{\pi}^{i+1} \leftarrow f(\overline{\pi}^i, L^i(\pi))$ based on some rule.
5: **end for**

- Caveat: In the IL problem, the objective $L(\pi, P^{\overline{\pi}^i}, \pi^{\mathsf{E}})$ is not revealed to the learner since the state-action distribution $P^{\overline{\pi}^i}$ is unknown.
- In each round i, we rollout $\overline{\pi}^i$ to collect a trajectory $(s_1^i, a_1^i, \cdots, s_H^i, a_H^i)$ and establish an empirical estimation $\widehat{P}^{\overline{\pi}^i}$, i.e., $\widehat{P}_h^{\overline{\pi}^i}(s, a) = \mathbb{I}\{(s_h^i, a_h^i) = (s, a)\}$.
- This is not a problem due to $L(\pi, P^{\overline{\pi}^i}, \pi^{\mathsf{E}}) = \mathbb{E}\left[L(\pi, \widehat{P}^{\overline{\pi}^i}, \pi^{\mathsf{E}}) | \overline{\pi}^i\right]$.

Online Learning Framework

Algorithm 2 Online Learning Framework

1: **Input:** Uniformly initialized policy $\overline{\pi}^1$
2: **for** $i = 1, 2, \cdots, m$ **do**
3: The learner takes policy $\overline{\pi}^i$ and observe objective function $L^i(\pi) = L(\pi, \widehat{P}^{\overline{\pi}^i}, \pi^{\mathsf{E}})$
4: The learner updates the policy $\overline{\pi}^{i+1} \leftarrow f(\overline{\pi}^i, L^i(\pi))$ based on mirror descent.
5: **end for**

- $L(\pi, \widehat{P}^{\overline{\pi}^i}, \pi^{\mathsf{E}})$ is linear w.r.t π and online mirror descent can be applied to solve this online linear optimization problem.
- Apply the online mirror descent theory [Shalev-Shwartz, 2012] with a little modification.

$$\sum_{i=1}^{m} L(\overline{\pi}^i, \widehat{P}^{\overline{\pi}^i}, \pi^{\mathsf{E}}) - \min_{\pi} \sum_{i=1}^{m} L(\pi, \widehat{P}^{\overline{\pi}^i}, \pi^{\mathsf{E}}) \leq \mathcal{O}\left(H|\mathcal{S}|\log(|\mathcal{A}|)\right).$$

- Modification: leverage $\min_\pi \sum_{i=1}^m L(\pi, \widehat{P}^{\overline{\pi}^i}, \pi^{\mathsf{E}}) = 0$ to obtain this constant regret.

Lower Bound Under the Active Setting

Theorem 2: Lower Bound Under the Active Setting

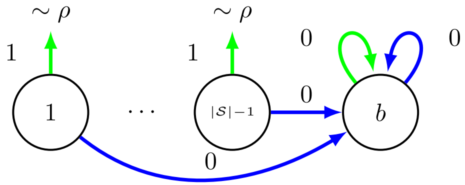
Under the active setting and μ-recoverability assumption, for any algorithm, there exists an IL problem such that

$$V(\pi^{\mathsf{E}}) - \mathbb{E}\left[V(\overline{\pi})\right] \gtrsim \frac{\mu|\mathcal{S}|H}{m}.$$

Here $\overline{\pi}$ is the output of the algorithm on this IL problem.

- The upper bound $\widetilde{\mathcal{O}}\left(\frac{\mu|\mathcal{S}|H}{m}\right)$ of the above algorithm nearly matches this lower bound, which implies that this algorithm is minimax optimal.

Proof of Lower Bound Under the Active Setting



Let $\mathcal{M}$ be the above MDP, which is used to establish the lower bound in the worst-case analysis. To satisfy the μ-recoverability assumption, we scale the reward by a factor of μ/H and the resultant MDP is denoted as $\mathcal{M}_\mu$.

$$V_{\mathcal{M}_\mu}(\pi^{\mathsf{E}}) - \mathbb{E}\left[V_{\mathcal{M}_\mu}(\overline{\pi})\right] = \frac{\mu}{H}\left(V_{\mathcal{M}}(\pi^{\mathsf{E}}) - \mathbb{E}\left[V_{\mathcal{M}}(\overline{\pi})\right]\right) \gtrsim \frac{\mu}{H}\frac{|\mathcal{S}|H^2}{m} = \frac{\mu|\mathcal{S}|H}{m}.$$

Here $\gtrsim$ follows the lower bound of $\Omega\left(\frac{|\mathcal{S}|H^2}{m}\right)$ in $\mathcal{M}$.

Lower Bound Under the Offline Setting

Theorem 3

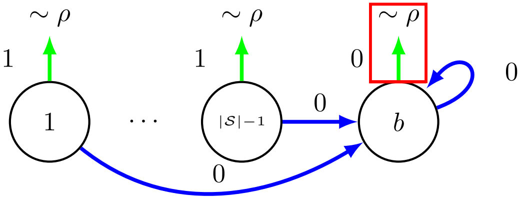
Under the offline setting and μ-recoverability assumption, for any algorithm, there exists an IL problem such that

$$V(\pi^{\mathsf{E}}) - \mathbb{E}\left[V(\overline{\pi})\right] \gtrsim \frac{|\mathcal{S}|H^2}{m}.$$

Here $\overline{\pi}$ is the output of the algorithm on this IL problem.

- Recall the minimax rate of $\frac{\mu|\mathcal{S}|H}{m}$ under the active setting, which provably shows the benefits of interactions with the MDP.
- This result establishes a clear separation between the policy value gap incurred by algorithms under the offline setting such as BC, and algorithms which can interact with the MDP.

The Hard Instance Under the Offline Setting



- At the bad state, they add a "recovery" action. By taking this recover action, the agent returns to good states.
- Due to the recovery action, this MDP satisfies μ-recoverability condition for any $\mu \geq 1$.
- Since the offline dataset does not cover the bad state, any offline IL algorithm fails to identify this recovery action with a probability of $1 - \frac{1}{|\mathcal{A}|}$ and thus suffers the same policy value gap as in the original MDP.

References I

I. Goodfellow, J. Pouget-Abadie, M. Mirza, B. Xu, D. Warde-Farley, S. Ozair, A. Courville, and Y. Bengio. Generative adversarial nets. Advances in neural information processing systems, 27, 2014.

J. Ho and S. Ermon. Generative adversarial imitation learning. In Advances in Neural Information Processing Systems 29, pages 4565–4573, 2016.

I. Kostrikov, K. K. Agrawal, D. Dwibedi, S. Levine, and J. Tompson. Discriminator-actor-critic: Addressing sample inefficiency and reward bias in adversarial imitation learning. In Proceedings of the 7th International Conference on Learning Representations, 2019.

D. Pomerleau. Efficient training of artificial neural networks for autonomous navigation. Neural Computation, 3(1):88–97, 1991.

References II

N. Rajaraman, L. F. Yang, J. Jiao, and K. Ramchandran. Toward the fundamental limits of imitation learning. arXiv, 2009.05990, 2020.

S. Ross and D. Bagnell. Efficient reductions for imitation learning. In Proceedings of the 13rd International Conference on Artificial Intelligence and Statistics, pages 661–668, 2010.

S. Ross and J. A. Bagnell. Reinforcement and imitation learning via interactive no-regret learning. arXiv preprint arXiv:1406.5979, 2014.

S. Ross, G. J. Gordon, and D. Bagnell. A reduction of imitation learning and structured prediction to no-regret online learning. In Proceedings of the 14th International Conference on Artificial Intelligence and Statistics, pages 627–635, 2011.

S. Shalev-Shwartz. Online learning and online convex optimization. Foundations and Trends in Machine Learning, 4(2):107–194, 2012.

References III

H. van Hasselt, A. Guez, and D. Silver. Deep reinforcement learning with double q-learning. In Proceedings of the Thirtieth AAAI Conference on Artificial Intelligence, February 12-17, 2016, Phoenix, Arizona, USA, pages 2094–2100. AAAI Press, 2016.