Provably Efficient Imitation Learning from Observations

Wen Sun    Anirudh Vemula    Byron Boots    J.Andrew Bagnell

Presented by: Zichu Liu
Motivation: Imitation Learning from Observations (ILFO)

Trajectories of Observations

Learning From Observations

No interactive expert, no expert action, no reset, no cost signals
Finite time horizon (T-step) Episodic MDP
Forward Adversarial Imitation Learning (FAIL)

\[ \pi_0(a|x_0), \pi_1(a|x_1), \ldots, \pi_T(a|x_T) \]
Decomposition into T subtasks

At time T, \( \{\pi_1, ..., \pi_{T-1}\} \) are learned already and fixed.

A state distribution \( \nu_T(x) \) is induced by \( \{\pi_1, ..., \pi_{T-1}\} \)

Expert policy \( \pi^* \) naturally induces a distribution \( \mu_T^* \)

We want to learn a policy \( \pi_T \in \Pi_T \) such that the resulting observation distribution from \( \{\pi_1, ..., \pi_{T-1}, \pi_T\} \) at time step T is close to the expert’s observation distribution \( \mu_T^* \) at time step T
Divergence: Integral Probability Metrics (IPM)

\[ d_{\mathcal{F}}(P_1, P_2) = \sup_{f \in \mathcal{F}} (\mathbb{E}_{x \sim P_1} [f(x)] - E_{x \sim P_2} [f(x)]) \]

- \( \mathcal{F} = \{ f : \|f\|_{\infty} \leq 1 \} \): Total Variation distance
- \( \mathcal{F} = \{ f : \|f\|_{L} \leq 1 \} \): Wasserstein distance
- \( \mathcal{F} = \{ f : \|f\|_{H} \leq 1 \} \): Maximum mean discrepancy
Learning the First Policy $\pi_0$

$\sim \mu^*_1(x)$

$\sum_{x_0,a_0} P(x_0) \pi_0(a_0|x_0) P(x|x_0,a_0)$

$\min_{\pi_0 \in \Pi} \max_{f \in F} f(\cdot) - f(\cdot)$
Learning the Second Policy $\pi_1$

$\sim \mu_2^*(x)$

$\sim \nu_2(x) = \sum_{x_1, a_1} \nu_1(x_1) \pi_1(a_1 | x_1) P(x | x_1, a_1)$

$\min_{\pi_1 \in \Pi} \max_{f \in \mathcal{F}} f(\cdot) - f(\cdot)$

Expert Distribution

Learner Distribution
Learning the Second Policy $\pi_1$

$\sim \mu_2^*(x)$

Expert Distribution

$Learner\ Distribution$

$\min_{\pi_1 \in \Pi} \max_{f \in \mathcal{F}} f(\pi_1) - f(\text{Expert})$

$\sim \nu_2(x) = \sum_{x_1, a_1} \nu_1(x_1) \pi_1(a_1 | x_1) P(x | x_1, a_1)$
Learning the Second Policy $\pi_1$

$\sim \mu_2^*(x)$

$\min_{\pi_1 \in \Pi} \max_{f \in \mathcal{F}} f(\pi_1(x)) - f(\pi_1(x))$

$\sim \nu_2(x) = \sum_{x_1, a_1} \nu_1(x_1) \pi_1(a_1|x_1) P(x|x_1, a_1)$

Expert Distribution

Learner Distribution
Learning the Third Policy $\pi_2$

$\sim \mu_3^*(x)$

$\min_{\pi_2 \in \Pi} \max_{f \in F} f(\cdot) - f(\cdot)$

$\sim \nu_3(x)$
Learning $\pi_T$

Given the distribution $\nu_T$ induced by $\{\pi_1, \cdots, \pi_T\} \in \Pi$, the observation distribution at time step $T + 1$ as

$$\nu_{T+1}(x) = \sum_{x_T, a_{T-1}} \nu_T(x_T) \pi(a_T | x_T) P(x | x_T, a_T)$$

Expert distribution at time step $T + 1$ is denoted as $\mu^*_{T+1}$

$\pi_T$ is obtained via minimizing the divergence between $\nu_{T+1}$ and $\mu^*_{T+1}$

$$\pi_T = \arg\min_{\pi \in \Pi} \max_{f \in F} f(\nu_{T+1}) - f(\mu^*_{T+1})$$
Learning $\pi_T$

However, the divergence $\max_{f \in \mathcal{F}} f(\nu_{T+1}) - f(\mu^*_{T+1})$ is not directly measurable since we do not have access to $\mu^*_{T+1}$ but only samples from $\mu^*_{T+1}$. 
Learning $\pi_T$

To estimate this divergence, we draw a dataset

$$\mathcal{D} = \{(x_T^i, a_T^i, x_{T+1}^i)\}$$

such that $x_T^i \sim \nu_T$, $a_T^i \sim \text{U}(A)$, $x_{T+1}^i \sim \text{P}(\cdot | x_T^i, a_T^i)$

Observations from expert $\mathcal{D}^* = \{\tilde{x}_{T+1}^i\}_{i=1}^{N'} \sim \mu_{T+1}^*$
Learning $\pi_T$

Empirical estimation of divergence:

$$\max_{f \in \mathcal{F}} \frac{K}{N} \sum_{i=1}^{N} \pi(a^i_T | x^i_T) f(x^i_{T+1}) - \frac{1}{N'} \sum_{i=1}^{N'} f(\tilde{x}^i_{T+1})$$

Where the importance weight $K\pi(a^i_T | x^i_T)$ is used to account for the fact that we draw actions uniformly from $A$ but want to evaluate $\pi$. 
Learning $\pi_T$

Now define the utility function of the two-player game:

$$u(\pi, f) = \frac{K}{N} \sum_{i=1}^{N} \pi(a_T^i|x_T^i) f(x_{T+1}^i) - \frac{1}{N'} \sum_{i=1}^{N'} f(\bar{x}_{T+1}^i)$$

Then we have the two-player game with solution $(\pi^*, f^*)$:

$$f^* = \arg \max_{f \in F} u(\pi^*, f)$$

$$\pi^* = \arg \min_{\pi \in \Pi} u(\pi, f^*)$$
Algorithm 1 Min-Max Game \((\mathcal{D}^*, \mathcal{D}, \Pi, \mathcal{F}, T)\)

1: Initialize \(\pi^0 \in \Pi\)
2: \textbf{for} \(n = 1\) to \(T\) \textbf{do}
3: \(f^n = \arg\max_{f \in \mathcal{F}} u(\pi^n, f)\) (LP Oracle)
4: \(u^n = u(\pi^n, f^n)\)
5: \(\pi^{n+1} = \arg\min_{\pi \in \Pi} \sum_{t=1}^{n} u(\pi, f^t) + \phi(\pi)\) (Regularized CS Oracle)
6: \textbf{end for}
7: \textbf{Output:} \(\pi^{n^*}\) with \(n^* = \arg\min_{n \in [T]} u^n\)
Algorithm 2 FAIL($\{\Pi_h\}_h$, $\{\mathcal{F}_h\}_h$, $\epsilon$, $n$, $n'$, $T$)

1: Set $\pi = \emptyset$
2: for $h = 1$ to $H - 1$ do
3: Extract expert’s data at $h + 1$: $\tilde{D} = \{\tilde{x}_{h+1}^i\}_{i=1}^{n'}$
4: $D = \emptyset$
5: for $i = 1$ to $n$ do
6: Reset $x_1^{(i)} \sim \rho$
7: Execute $\pi = \{\pi_1, \ldots, \pi_{h-1}\}$ to generate state $x_h^i$
8: Execute $a_h^i \sim U(\mathcal{A})$ to generate $x_{h+1}^i$ and add $(x_h^i, a_h^i, x_{h+1}^i)$ to $D$
9: end for
10: Set $\pi_h$ to be the return of Algorithm 1 with inputs $(\tilde{D}, D, \Pi_h, \mathcal{F}_{h+1}, T)$
11: Append $\pi_h$ to $\pi$
12: end for
Assumption: (Realizability and Capacity of Function Class)

Assume $\Pi$ and $\mathcal{F}$ are finite and contain $\pi_t^*$ and $f_t^*$, i.e.,

$$
\pi_t^* \in \Pi \text{ and } f_t^* \in \mathcal{F}, \forall t \in [0,T]
$$
Convergence Result

**Theorem 3.1.** Given $\epsilon \in (0, 1]$, $\delta \in (0, 1]$, set $T = \Theta \left( \frac{4K^2}{\epsilon^2} \right)$, $N = N' = \Theta \left( \frac{K \log(||\Pi_h||F_{h+1}|/\delta)}{\epsilon^2} \right)$, Algorithm 1 outputs $\pi$ such that with probability at least $1 - \delta$,

$$
\left| d_{F_{h+1}}(\pi|\nu_h, \mu_{h+1}^*) - \min_{\pi' \in \Pi_h} d_{F_{h+1}}(\pi'|\nu_h, \mu_{h+1}^*) \right| \leq O(\epsilon).
$$
Convergence Result

Given $\epsilon \in (0,1]$, $\delta \in (0,1]$, algorithm 1 outputs $\pi$ such that with probability $1 - \delta$,

$$\left| \left\{ \max_{f} f(\nu_{T+1}) - f(\mu_{T+1}^*) \right\} - \left\{ \min_{\pi'} \max_{f} f(\nu_{T+1}) - f(\mu_{T+1}^*) \right\} \right| < O(\epsilon)$$

$$\left| \text{Div}(\pi) - \min_{\pi'} \text{Div}(\pi') \right| < O(\epsilon)$$
Simulation:

<table>
<thead>
<tr>
<th>Model</th>
<th>T</th>
<th>Dense/Sparse Reward Task</th>
</tr>
</thead>
<tbody>
<tr>
<td>Swimmer</td>
<td>100</td>
<td>Dense</td>
</tr>
<tr>
<td>Reacher</td>
<td>50</td>
<td>Dense/Sparse</td>
</tr>
<tr>
<td>FetchReach</td>
<td>50</td>
<td>Sparse</td>
</tr>
</tbody>
</table>
Simulation:

Compare FAIL with modified GAIL:

The modified version of GAIL uses RL methods to minimize the divergence between the learner’s average state distribution and expert’s average state distribution.
### Experiment (Dense Reward)

**SwimmerDiscrete**

<table>
<thead>
<tr>
<th>Number of Expert Trajectories</th>
<th>Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Expert: 35</td>
</tr>
<tr>
<td></td>
<td>FAIL: 30</td>
</tr>
<tr>
<td></td>
<td>GAIL: 25</td>
</tr>
</tbody>
</table>

**ReacherDiscrete**

<table>
<thead>
<tr>
<th>Number of Expert Trajectories</th>
<th>Return</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>Expert: -7.50</td>
</tr>
<tr>
<td></td>
<td>FAIL: -8.00</td>
</tr>
<tr>
<td></td>
<td>GAIL: -8.25</td>
</tr>
</tbody>
</table>
Experiment (Dense Reward)
Summary

• This paper point out a new direction of imitation learning research: imitation learning from observation alone. (ILFO)
• Propose FAIL, an algorithm that is theoretically guaranteed to solve the ILFO problems.
• Modify GAIL to solve ILFO problem, experimentally demonstrate that GAIL and FAIL work equivalently well in problems with dense reward, and FAIL outperforms GAIL on sparse reward MDPs.