Adversarially Robust Imitation Learning

Anonymous Author(s)
Affiliation
Address
email

A Proof of Theorem 4.1

Proof. Since Alg. 1 runs no-regret online learner to update π on the sequence of loss functions $\{\ell_t(\pi)\}$, we must have:

$$\sum_{t=0}^{T} \ell_t(\pi_t) - \min_{\pi \in \Pi} \sum_{t=0}^{T} \ell_t(\pi) \leq o(T). \tag{1}$$

Add and subtract $\sum_{t=0}^{T} \ell_t(\pi^e)$ on the left hand side of the above inequality, we have:

$$\sum_{t=0}^{T} \ell_t(\pi_t) - \sum_{t=0}^{T} \ell_t(\pi^e) \leq o(T) + \left(\min_{\pi \in \Pi} \sum_{t=0}^{T} \ell_t(\pi) - \sum_{t=0}^{T} \ell_t(\pi^e) \right) \tag{2}$$

Since we operate under the realizability setting, the term inside the parenthesis on the RHS of the above inequality is guaranteed to be less than or equal to zero. Hence, the above inequality simplifies to:

$$\sum_{t=0}^{T} \ell_t(\pi_t) \leq \sum_{t=0}^{T} \ell_t(\pi^e) + o(T). \tag{3}$$

For $\ell_t(\pi^e)$, using the definition of ℓ_t, we see:

$$\ell_t(\pi^e) = \mathbb{E}_{s \sim d_{\pi_t \circ f_t}} \left[\mathbb{E}_{a \sim \pi^e(f_t(s))} [\ell(a, \pi^e(s))] \right] = \mathbb{E}_{s \sim d_{\pi_t \circ f_t}} \left[\ell(\pi^e(f_t(s)), \pi^e(s)) \right] = 0, \tag{4}$$

where we use the expert robustness definition above (i.e., $\pi^e(f(s)) = \pi^e(s)$ for all s and f). Hence, we have:

$$\sum_{t=0}^{T} \ell_t(\pi_t) \leq T\epsilon_e + o(T). \tag{5}$$

Now we need to lower bound $\ell_t(\pi)$. Using the definition of ℓ_t again, we have:

$$\max_f \mathcal{L}(\pi_{i^\star}, f) \leq \mathcal{L}(\pi_{i^\star}, f_{i^\star}) + \epsilon_{rl} \leq \frac{1}{T} \sum_t \mathcal{L}(\pi_t, f_t) + \epsilon_{rl} = \frac{1}{T} \sum_t \ell_t(\pi_t) + \epsilon_{rl}, \tag{6}$$

where the first inequality comes from the assumption that the RL solver returns a ϵ_{rl} near-optimal solution.

Combine all the results above together, we get:

$$\max_f \mathcal{L}(\pi_{i^\star}, f) \leq \epsilon_{rl} + \epsilon_e + o(T)/T. \tag{7}$$

Hence as T approaches to ∞, we have that the long-term prediction loss of the learned policy $\pi_{i^\star}$ under the worst possible adversarial attack from $\mathcal{F}$ is upper bounded by ϵ_{rl}—the error introduced from optimizing f_t.

Under agnostic setting, it is not guaranteed that there exists $a = \pi^e(s)$, where $a \sim \pi(s)$ and $\pi \in \Pi$ for $\forall s \in \mathcal{S}$. However, it can be assumed that the error between a and $\pi^e(s)$ is bounded by a small number ϵ_a, namely $\mathbb{E}_{a \sim \pi(s)}[a \neq \pi^e(s)] \leq \epsilon_a$. Hence Equation 4 is modified as:

$$\ell_t(\pi^e) = \mathbb{E}_{s \sim d_{\pi_t \circ f_t}} \left[\mathbb{E}_{a \sim \pi^e(f_t(s))} [\ell(a, \pi^e(s))] \right] \leq \epsilon_a. \tag{8}$$

Submitted to the 5th Conference on Robot Learning (CoRL 2021). Do not distribute.

Plugging this to Equation 3, we have:

$$\sum_{t=0}^{T} \ell_t(\pi_t) \leq T(\epsilon_e + \epsilon_a) + o(T). \tag{9}$$

Combining the results with Equation 6, we get

$$\max_f \mathcal{L}(\pi_{i^\star}, f) \leq \epsilon_{rl} + \epsilon_e + \epsilon_a + o(T)/T. \tag{10}$$

Similarly to the realizability setting, as T approaches to ∞, the long-term prediction loss of the learned policy $\pi_{i^\star}$ under the worst possible adversarial attack from $\mathcal{F}$ is still upper bounded.

□

B Experiments in Detail

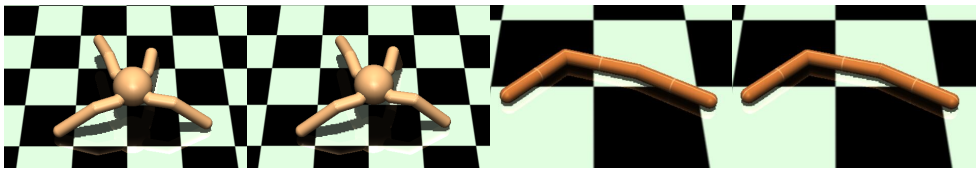
B.0.1 Experiment Setting

Table 1 shows ϵ in each settings.

And as figure 1, the perturbation added to each state is bounded to a small value such that it's imperceptible but can lead to significant performance decrease for the student network before trained by ARIL.

Table 1: ϵ for Each Attack

Environments	Ant-v2	HalfCheetah-v2	Swimmer-v2
sensory IL	0.5	0.04	0.25
physical	0.01	0.024	0.014



(a) Ant ($\epsilon = 0.1$) (b) Ant ($\epsilon = 0.1$) (c) Swimmer ($\epsilon = 0.25$) (d) Swimmer ($\epsilon = 0.25$)

Figure 1: Robot observation before (a,c) and after (b,d) attack.

B.0.2 Training Details

Both attacker and student are trained using Adam optimizer at learning rate of 0.001.

At the start of ARIL algorithm, we initialized the buffer for DAgger algorithm with size $5M$ timesteps. In each attack stage, it takes $1e6$ timesteps for the attacker to collect the trajectories each time. Then at each defense stage, we collected 40 trajectories of student under attack and labeled them with expert actions into the DAgger buffer.

Table 2: Hyperparameters for training attacker

entropy loss coefficient	0.01
value loss coefficient	0.5
clip range	0.2