Robust Imitation Learning against Variations in Environment Dynamics

Jongseong Chae[1], Seungyul Han[2], Whiyoung Jung[1],
Myungsik Cho[1], Sungho Choi[1], Youngchul Sung[1]

1 School of Electrical Engineering, KAIST
2 Artificial Intelligence Graduate School, UNIST

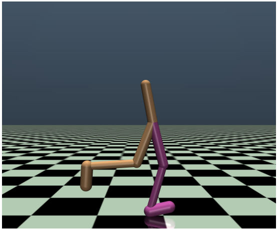
ICML 2022

Imitation Learning

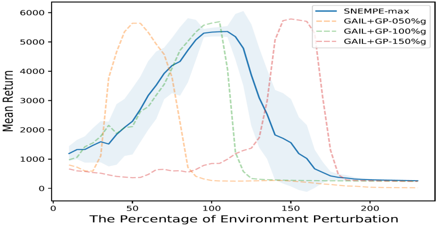
- Reward function design: it may be difficult to make a reward function for successful application of RL
- IL learns a policy from an Expert's Demonstration $\tau_E = (s_0, a_0, s_1, a_1, \cdots)$
- Previous methods: Behavior Cloning, GAIL, etc.

Robust Imitation Learning

- Robustness: the underlying dynamics are highly likely to be perturbed in the real world
- We need a Robust IL framework that can perform well in various environments with different dynamics by using expert demonstrations
  - ✓ For example, $\tau_E^{rainy\ day}$, $\tau_E^{clear\ day}$, $\tau_E^{snowy\ day}$



(a) Walker2d



(b) IL Performance with $\tau_E^{0.5g}$, $\tau_E^{1.0g}$, $\tau_E^{1.5g}$

- Robust RL

$$\max_{\pi} \min_{\mathcal{P}^i \in P} \mathbb{E}_{\pi}[G_t | \mathcal{P}^i]$$

- An IL algorithm that is trained in a single environment and uses multiple expert demonstrations ($\tau_E^{0.5g}$, $\tau_E^{1.0g}$, $\tau_E^{1.5g}$)

$$\min_{\pi} \max\{D_1(\tau_E^{0.5g}, \tau_\pi), D_2(\tau_E^{1.0g}, \tau_\pi), D_3(\tau_E^{1.5g}, \tau_\pi)\}$$

$\rightarrow$ Policy interaction with the single environment is not enough to handle the dynamics variation even with multiple expert demonstrations

- **Setup**: An MDP collection $\mathcal{C} = \{\mathcal{M} =< \mathcal{S}, \mathcal{A}, \mathcal{P}_\zeta, r, \gamma >,\ \zeta \in Z\}$
  - ✓ Transition probability $\mathcal{P}_\zeta$ modeling the dynamics is parameterized by dynamics parameter ζ which is in a continuous parameter space
  - ✓ $\mathcal{S}$ and $\mathcal{A}$ are the same for all members of $\mathcal{C}$
  - ✓ Reward function r is not available
- **Goal**: To learn a policy π that performs well for all members in the MDP collection $\mathcal{C}$

- N MDPs with dynamics $\mathcal{P}_{\zeta_1}, \cdots, \mathcal{P}_{\zeta_N}$ are sampled among $\mathcal{C}$
- The sampled environments are for both *policy interaction* and *expert demonstrations*

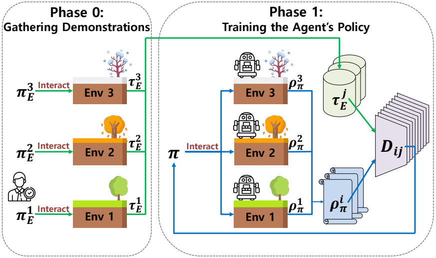


Figure: Overall Structure

Simple Approach: Occupancy Measure Matching

In a Single Environment

$$\rho_\pi(s,a) = \mu_0(s)\pi(a|s) + \gamma \int_{(s',a')} \mathcal{P}(s|s',a')\rho_\pi(s',a')\pi(a|s)$$

✓ The Bellman flow constraint has the unique solution ρ_π

→ There is a 1-to-1 correspondence between π and ρ

→ We can seek a policy π close to the expert policy π_E by using the occupancy measure matching technique that is used in GAIL

In Multiple Environments

$$\rho_\pi(s,a) = \mu_0(s)\pi(a|s) + \frac{\gamma}{N}\sum_{i=1}^{N} \int_{(s',a')} \mathcal{P}_{\zeta_i}(s|s',a')\rho^i_\pi(s',a')\pi(a|s)$$

✓ There exist many solutions, so $\rho_\pi = \frac{1}{N}\sum_{i=1}^{N} \rho^i_\pi$ can be many

→ The relation between π and ρ can be 1-to-many

The Proposed Robust Imitation Learning Framework

An Objective Function not requiring Occupancy Measures:

$$\min_{\pi} \mathbb{E}_{s \sim \frac{1}{N} \sum_{i=1}^{N} \mu_{\pi}^{i}} \left[\sum_{j=1}^{N} \lambda_j(s) \cdot \mathcal{D}(\pi(\cdot|s), \pi_E^j(\cdot|s)) \right] \quad (1)$$

- ✓ $\lambda_j(s)$ is the weight to determine how much $\pi_E^j(\cdot|s)$ is imitated, $\mathcal{D}$ is a divergence between two policy distributions
- ✓ However, (1) requires the expert policies π_E^j which are not available

Theorem (Practical Objective Function):

$$\min_{\pi} \sum_{i=1}^{N} \sum_{j=1}^{N} \max_{D_{ij}} \left\{ \mathbb{E}_{(s,a)\sim\rho_\pi^i} \left[\lambda_j(s) \log(1 - D_{ij}(s,a))\right] + \mathbb{E}_{(s,a)\sim\rho_E^j} \left[\frac{\mu_\pi^i(s)}{\mu_E^j(s)} \lambda_j(s) \log(D_{ij}(s,a)) \right] \right\} \quad (2)$$

- ✓ D_{ij} is a discriminator that distinguishes whether (s,a) is from policy π interacting with i-th sampled environment or from j-th expert π_E^j
- ✓ (2) requires expert demonstrations $\tau_E^j \sim \rho_E^j$ not expert policies π_E^j

Experiments: Baseline Algorithms

- Even without guarantee of the recovery of policy from occupancy measure, we can apply the occupancy meausure matching technique to the multiple environments setting
- We compared our algorithm with the following baseline algorithms
  - OMME (closest to our algorithm)

$$\min_{\pi} \sum_{j=1}^{N} \lambda_j \mathcal{D}_{JS}(\bar{\rho}_{\pi}, \bar{\rho}_E^j)$$

  - GAIL-mixture

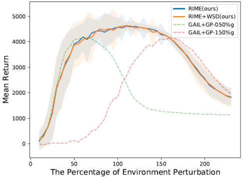
$$\min_{\pi} \mathcal{D}_{JS}(\sum_{i=1}^{N} \bar{\rho}_{\pi}^i / N, \sum_{j=1}^{N} \bar{\rho}_E^j / N)$$

  - GAIL-single

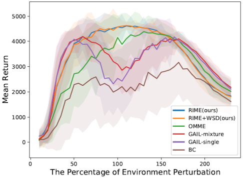
$$\min_{\pi} \sum_{i=1}^{N} \mathcal{D}_{JS}(\bar{\rho}_{\pi}^i, \bar{\rho}_E^i)$$

Experiments: 1-D Perturbation Case

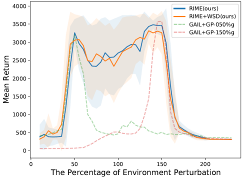
- MuJoCo tasks with 1-D dynamics perturbation (gravity or mass)
  → Our algorithm with $N = 2$ sampled environments $(50\%\zeta_0,\ 150\%\zeta_0)$ is robust over the dynamics variation between the sampled dynamics.



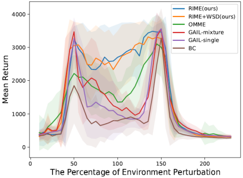
(a) Ant+Gravity: performance



(b) Ant+Gravity: comparisons



(c) Hopper+Gravity: performance



(d) Hopper+Gravity: comparisons

Figure: Performance for our algorithm and baseline algorithms for MuJoCo tasks

Experiments: 2-D Perturbation Case

- MuJoCo tasks with 2-D dynamics perturbation (gravity and mass)
  $\rightarrow$ Our algorithm performs well within the joint gravity-mass dynamics parameter space by only sampling the four corner points.

Algorithm	Hopper + (G&M)	Walker2d + (G&M)	HalfCheetah + (G&M)	Ant + (G&M)
RIME (ours)	**3043.3 / 2430.8**	4463.4 / 3824.1	**3721.3** / 2753.1	**4671.7 / 4233.5**
RIME+WSD (ours)	2936.9 / **2331.6**	**4646.4 / 4000.2**	**3717.9 / 2891.7**	**4651.4 / 4304.5**
OMME	2573.4 / 1986.4	4488.8 / 3029.3	3498.5 / 2502.2	**4625.3** / 3594.5
GAIL-mixture	1636.4 / 712.0	3907.8 / 1245.1	3018.6 / 1982.3	3994.8 / 2746.1
GAIL-single	1684.9 / 840.0	3844.8 / 2484.2	3199.1 / 2072.6	3799.7 / 2194.1
BC	500.2 / 317.2	330.0 / 211.0	1289.3 / 30.2	1728.2 / 1032.7

Figure: The robustness performance of all algorithms

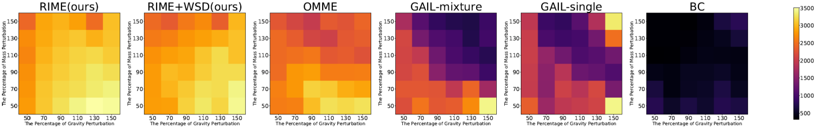
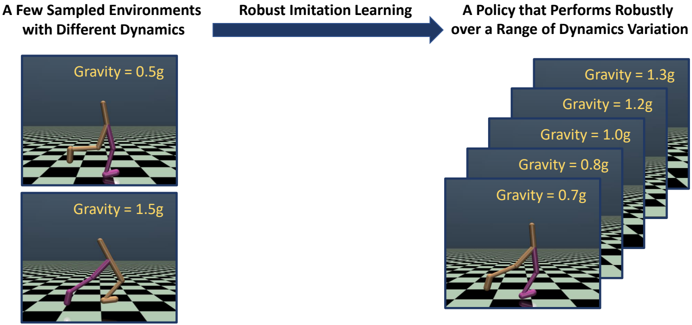


Figure: Performance for our algorithm and baseline algorithms for Hopper task

Conclusion

- In this paper, we have considered how to improve the robustness of IL to address both robustness and reward function design
- We propose a robust IL framework based on a few environments with sampled dynamics parameters
- Our proposed IL algorithm shows superior performance in robustness over the dynamics variation compared to the conventional IL baselines



Thank you!