# Generative Adversarial Imitation Learning

Stoyan Dimitrov

July 13, 2021

IMS, University of Stuttgart

# Table of contents

# The paper

## Main concern

Generative Adversarial Imitation Learning (Ho and Ermon, 2016):

- *IRL learns a cost function, which explains expert behavior ... but does not directly tell the learner how to act*
- *... - why, then, must we learn a cost function, if doing so possibly incurs significant computational expense yet fails to directly yield actions?*

# Starting point

Maximum causal entropy IRL:

$$\underset{c \in \mathcal{C}}{\text{maximize}} \left( \min_{\pi \in \Pi} -H(\pi) + \mathbb{E}_{\pi}[c(s,a)] \right) - \mathbb{E}_{\pi_E}[c(s,a)] \tag{1}$$

Start we some general $\psi$:

$$\mathrm{IRL}_{\psi}(\pi_E) = \underset{c \in \mathbb{R}^{\mathcal{S} \times \mathcal{A}}}{\arg\max} -\psi(c) + \left( \min_{\pi \in \Pi} -H(\pi) + \mathbb{E}_{\pi}[c(s,a)] \right) - \mathbb{E}_{\pi_E}[c(s,a)] \tag{3}$$

# $\psi$ as distance measure

- Skipped here: policy and its occupancy measure $\rho_\pi$ can be used interchangeably
- Proposition 3.2:

$$\mathrm{RL} \circ \mathrm{IRL}_\psi(\pi_E) = \arg\min_{\pi \in \Pi} -H(\pi) + \psi^*(\rho_\pi - \rho_{\pi_E}) \tag{4}$$

- How does the convex conjugate helps us:

$$\begin{aligned}\pi_A &\in \arg\min_\pi -H(\pi) + \psi^*(\rho_\pi - \rho_{\pi_E}) \\ &= \arg\min_\pi \max_c -H(\pi) - \psi(c) + \sum_{s,a} (\rho_\pi(s,a) - \rho_{\pi_E}(s,a))c(s,a)\end{aligned}$$

# Choices for $\psi$

- If $\psi$ is constant, the solution of (4) is simply matching the occupancy measure:

$$\underset{\rho\in\mathcal{D}}{\text{minimize}} -\bar{H}(\rho) \quad \text{subject to} \quad \rho(s,a)=\rho_E(s,a) \quad \forall\, s\in\mathcal{S}, a\in\mathcal{A} \tag{7}$$

- If $\psi$ is the indicator function $\delta_c$, the with $\delta_c^*(\rho_\pi - \rho_{\pi_E})$:

$$\underset{\pi}{\text{minimize}} -H(\pi) + \max_{c\in\mathcal{C}} \mathbb{E}_\pi[c(s,a)] - \mathbb{E}_{\pi_E}[c(s,a)] \tag{11}$$

# $\psi$ for GAIL

Proposed $\psi$:

$$\psi_{\mathrm{GA}}(c) \triangleq \begin{cases} \mathbb{E}_{\pi_E}[g(c(s,a))] & \text{if } c < 0 \\ +\infty & \text{otherwise} \end{cases} \quad \text{where } g(x) = \begin{cases} -x - \log(1 - e^x) & \text{if } x < 0 \\ +\infty & \text{otherwise} \end{cases} \tag{13}$$

... motivated by the form of $\psi^*$:

$$\psi^*_{\mathrm{GA}}(\rho_\pi - \rho_{\pi_E}) = \max_{D \in (0,1)^{\mathcal{S} \times \mathcal{A}}} \mathbb{E}_\pi[\log(D(s,a))] + \mathbb{E}_{\pi_E}[\log(1 - D(s,a))] \tag{14}$$

# The Algorithm

**Algorithm 1** Generative adversarial imitation learning

1: **Input:** Expert trajectories $\tau_E \sim \pi_E$, initial policy and discriminator parameters $\theta_0, w_0$
2: **for** $i = 0, 1, 2, \dots$ **do**
3: Sample trajectories $\tau_i \sim \pi_{\theta_i}$
4: Update the discriminator parameters from $w_i$ to $w_{i+1}$ with the gradient

$$\hat{\mathbb{E}}_{\tau_i}[\nabla_w \log(D_w(s,a))] + \hat{\mathbb{E}}_{\tau_E}[\nabla_w \log(1 - D_w(s,a))] \quad (17)$$

5: Take a policy step from $\theta_i$ to $\theta_{i+1}$, using the TRPO rule with cost function $\log(D_{w_{i+1}}(s,a))$. Specifically, take a KL-constrained natural gradient step with

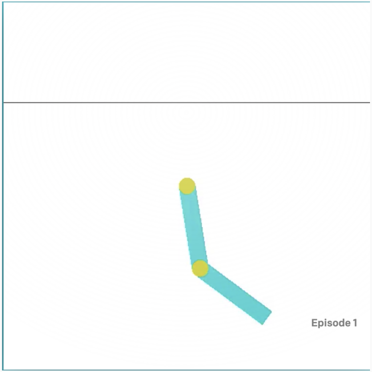
$$\begin{aligned} &\hat{\mathbb{E}}_{\tau_i}\left[\nabla_\theta \log \pi_\theta(a|s) Q(s,a)\right] - \lambda \nabla_\theta H(\pi_\theta), \\ &\text{where } Q(\bar{s},\bar{a}) = \hat{\mathbb{E}}_{\tau_i}[\log(D_{w_{i+1}}(s,a)) \mid s_0 = \bar{s}, a_0 = \bar{a}] \end{aligned} \quad (18)$$

6: **end for**

# Implementation

# The Environment

Acrobot-v1 - swing the lower end to a certain height:



# Components

- Generator - an actor-critic model for learning the policy, trained with PPO
- Discriminator - MLP model, trained with Binary Cross Entropy
- Expert policy - a generator object, trained with the original reward

# Hyperparameters and evaluation

- Expert trained for 50 epochs à 4000 steps
- All MLP with hidden size of 100 and Tanh activations
- 1, 4, 7 or 10 expert trajectories in the dataset
- 300 training iterations with 5000 steps/interactions

**Performance measure**: episode length

# Results

# Episode length per expert dataset size

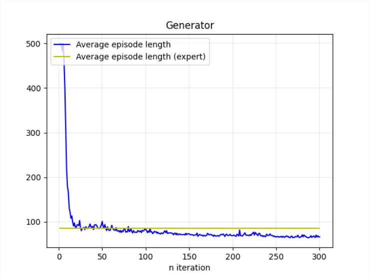



**Figure 1:** n=10

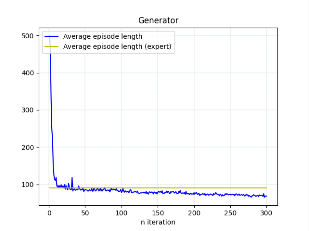



**Figure 2:** n=7

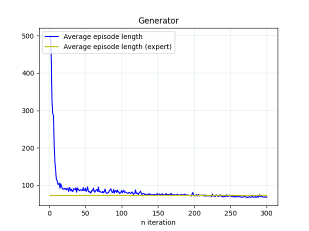



**Figure 3:** n=1

# Discriminator performance

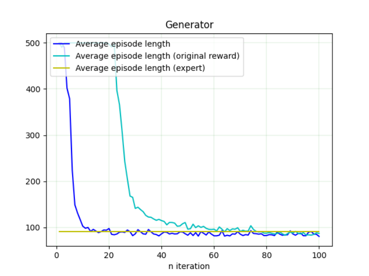



**Figure 4:** Discriminator vs. original reward

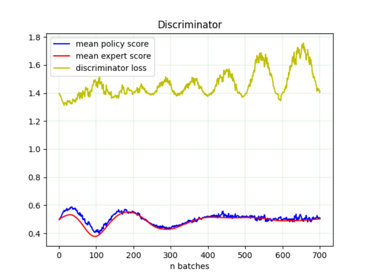



**Figure 5:** Average score for policy and expert samples

# References


Goodfellow, I., Pouget-Abadie, J., Mirza, M., Xu, B., Warde-Farley, D., Ozair, S., Courville, A., and Bengio, Y. (2014). Generative adversarial nets. In *Neural Information Processing Systems (NIPS)*.

Ho, J. and Ermon, S. (2016). Generative adversarial imitation learning. In *30th Annual Conference on Neural Information Processing Systems*, pages 4565–4573.

Mainprice, J. (2020). Reinforcement learning. Lectures in Reinforcement Learning in Summer semester 2020.

Ng, A. Y., Harada, D., and Russell, S. J. (1999). Policy invariance under reward transformations: Theory and application to reward shaping. In *Proceedings of the Sixteenth International Conference on Machine Learning*, ICML '99, page 278–287, San Francisco, CA, USA. Morgan Kaufmann Publishers Inc.

Ng, A. Y. and Russell, S. J. (2000). Algorithms for inverse reinforcement learning. In *Proceedings of the Seventeenth International Conference on Machine Learning*, pages 663–670.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement Learning: An Introduction*. MIT Press, Cambridge, Massachusetts.

Zheng, Z., Oh, J., and Singh, S. (2018). On learning intrinsic rewards for policy gradient methods. In *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, NIPS'18, page 4649–4659, Red Hook, NY, USA. Curran Associates Inc.