Generative Adversarial Imitation Learning

Jonathan Ho
Stanford University
hoj@cs.stanford.edu

Stefano Ermon
Stanford University
ermon@cs.stanford.edu

NIPS-2016

Outline

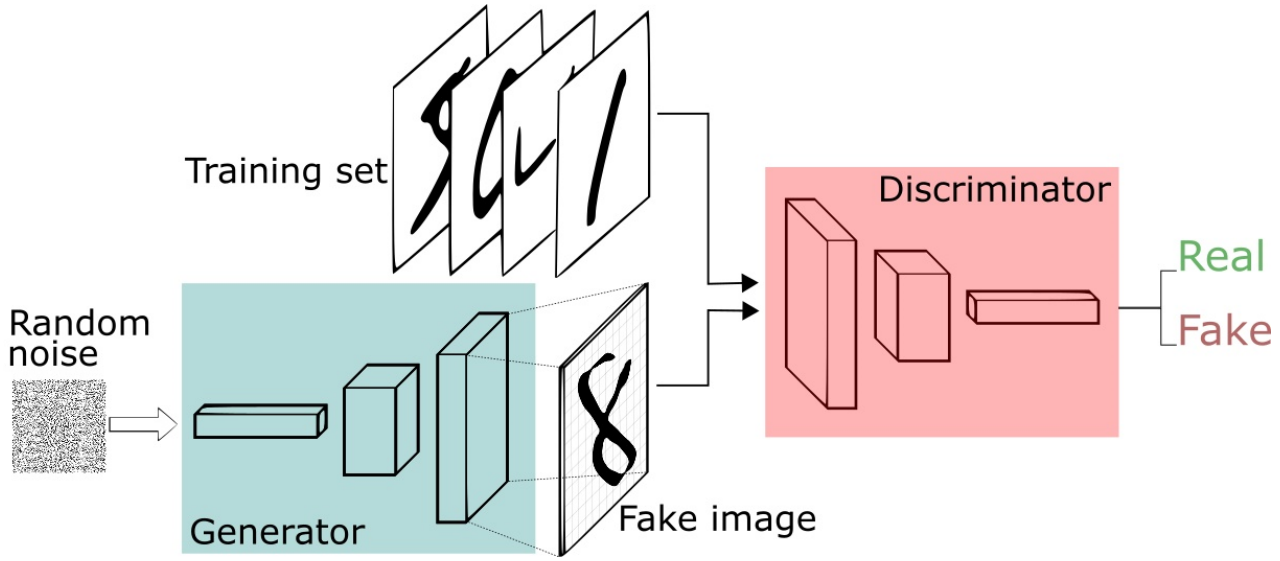
- Imitation Learning
- GAN
- Idea of GAIL
- Policy Gradient
- Algorithm
- Experiment

Imitation Learning

- Given a policy, we can roll out a trajectory (sequence of station-action pairs) by starting from an initial state and following the policy.
- What is imitation learning? Given many trajectories provided by human, learn a good policy.

GAN

- Goal of GAN: generate examples as realistic as possible
- Two components: Generator and Discriminator
  - Goal of Generator: generate realistic data.
  - Goal of Discriminator: distinguish data produced by generator from true data distribution



$$\min_G \max_D V(D,G) = \mathbb{E}_{\boldsymbol{x}\sim p_{\text{data}}(\boldsymbol{x})}[\log D(\boldsymbol{x})] + \mathbb{E}_{\boldsymbol{z}\sim p_{\boldsymbol{z}}(\boldsymbol{z})}[\log(1 - D(G(\boldsymbol{z})))].$$

Idea of GAIL

- Idea: combine GAN with IL
- In GAN generator improve by contesting with discriminator. In imitation learning we can also let policy contesting with discriminator. If Discriminator can’t distinguish whether the trajectories are provided by human, then the policy we learned are good enough.

Policy gradient

- In reinforcement learning our goal is to learn a good policy, but how to judge the goodness of a policy? The sum of reward of all the trajectories is largest.
- $R_\theta = \sum_\tau R(\tau)P(\tau|\theta)$
- $\nabla R_\theta = \sum_\tau R(\tau)\nabla P(\tau|\theta) = \sum_\tau R(\tau)P(\tau|\theta)\frac{\nabla P(\tau|\theta)}{P(\tau|\theta)} = \sum_\tau R(\tau)P(\tau|\theta)\nabla \log P(\tau|\theta)$
- $\nabla R_\theta \approx \frac{1}{m}\sum_{i=1}^{m} R(\tau)\,\nabla \log P(\tau|\theta)$

  $= \frac{1}{m}\sum_{i=1}^{m} R(\tau)\sum_{i=1}^{T}\nabla \log P(a_t|s_t,\theta)$

Algorithm

Algorithm 1 Generative adversarial imitation learning

1: **Input:** Expert trajectories $\tau_E \sim \pi_E$, initial policy and discriminator parameters θ_0, w_0
2: **for** $i = 0, 1, 2, \ldots$ **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

gradient of Discriminator

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

5: Take a policy step from θ_i to θ_{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

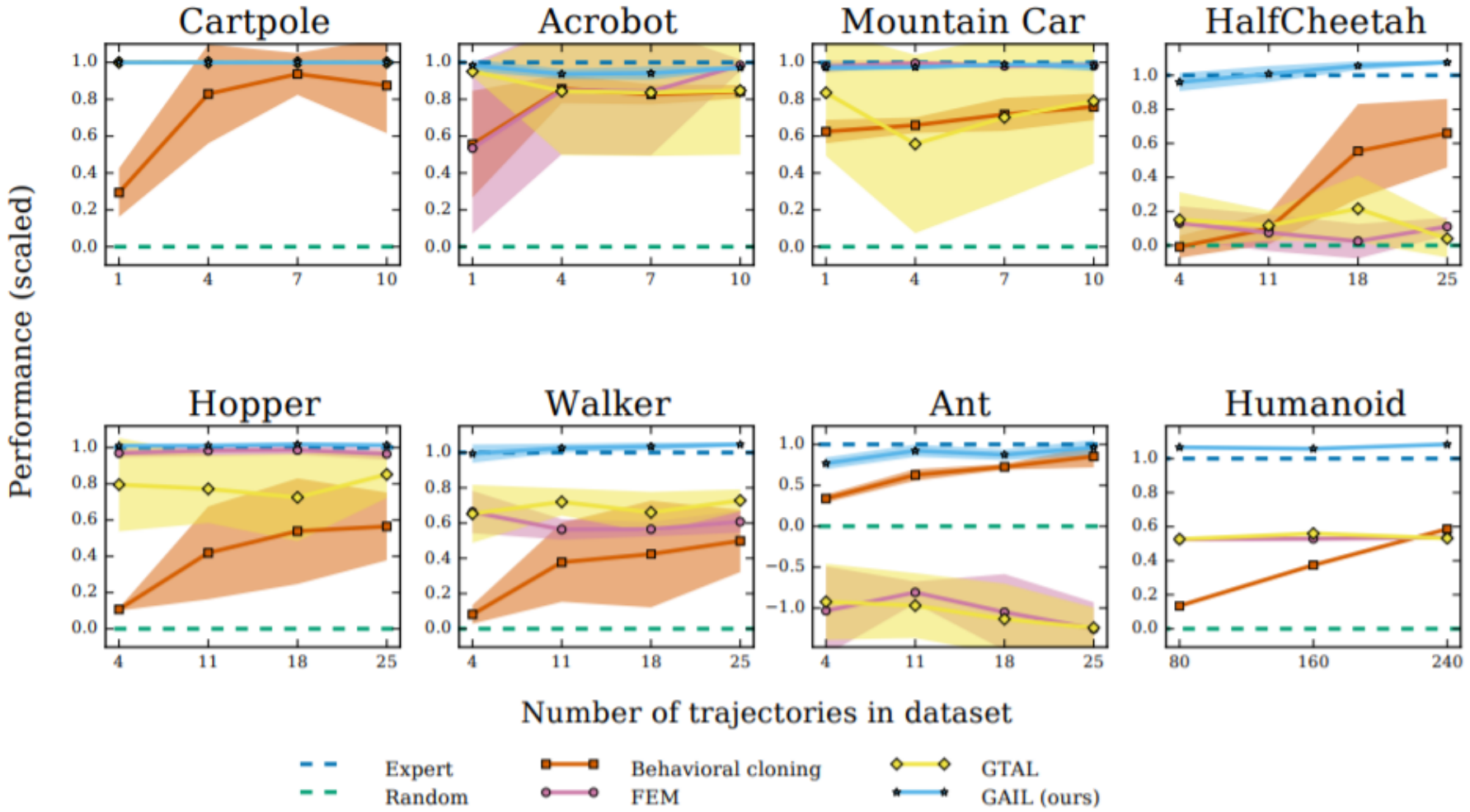
gradient of policy, Change cost to the cost of generator

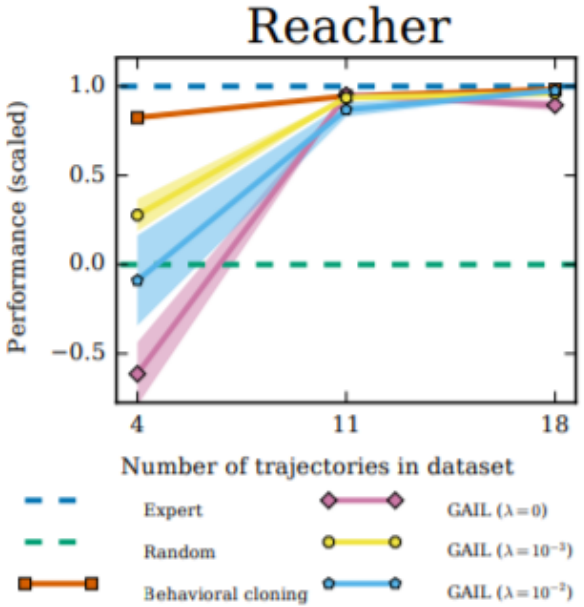
$$\hat{\mathbb{E}}_{\tau_i}[\nabla_\theta \log \pi_\theta(a|s) Q(s,a)] - \lambda \nabla_\theta H(\pi_\theta), \quad \text{where } Q(\bar{s},\bar{a}) = \hat{\mathbb{E}}_{\tau_i}[\log(D_{w_{i+1}}(s,a)) \,|\, s_0 = \bar{s}, a_0 = \bar{a}] \tag{18}$$

H: Entropy

6: **end for**

Experiment


Cartpole
Acrobot
Mountain Car
HalfCheetah
Hopper
Walker
Ant
Humanoid
Performance (scaled)
Number of trajectories in dataset
Expert
Random
Behavioral cloning
FEM
GTAL
GAIL (ours)


Reacher
Performance (scaled)
Number of trajectories in dataset
Expert
Random
Behavioral cloning
GAIL (λ=0)
GAIL (λ=10⁻²)
GAIL (λ=10⁻⁴)