Generative Adversarial Imitation Learning

Paper 9: Ho and Ermon, Generative Adversarial Imitation Learning, NIPS 2016

Giulia Ghisolfi, g.ghisolfi@studenti.unipi.it, 664222

Master Degree in Computer in Science (Artificial Intelligence Curriculum), University of Pisa
Intelligent Systems for Pattern Recognition course (760AA)
Academic Year: 2022-2023

Introduction

The paper focuses on a specific scenario of imitation learning where the learner receives expert demonstrations in the form of trajectory samples to perform a task. The learner is not allowed to request additional data from the expert during training and does not receive any form of reinforcement signal.

Main approaches suitable for this setting include:

- Behavioral cloning: This approach is relatively straightforward but typically requires a large amount of data to achieve good results.
- Inverse Reinforcement Learning (IRL) algorithms: These algorithms can be effective but are computationally expensive as they involve running reinforcement learning in an inner loop.

The purpose of the authors is to develop an algorithm that explicitly instructs us on how to act by directly learning a policy.

Model description

The model introducted in the paper, **Generative Adversarial Imitation Learning**, addresses these challenges by introducing a competition between two neural networks: a generator and a discriminator.

During training, the generator and discriminator engage in an **adversarial process**.

The goal of the **generator** is to imitate expert behavior by generating realistic actions.

On the other hand, the **discriminator** aims to minimize the classification error between expert actions and actions generated by the generator.

Model description

The **GAIL** algorithm follows an **iterative training procedure** where the generator and discriminator are alternately updated to improve their performance with respect to their objectives. Error backpropagation is utilized to update the weights of the generator and discriminator neural networks during training.

An important feature of GAIL is the use of a conditional architecture for the generator, allowing it to consider current observations and generate context-based actions. This enables the agent to make informed decisions based on the environment.

Key catch of the model

The **GAIL** algorithm is designed to **solve an imitation learning problem**, which can be **formulated as an optimization problem**. GAIL uses an iterative procedure to train the generator and discriminator to improve their performance with respect to the objective.

The problem can be described by

$$\min_{\pi} \varphi_{GA}(\rho_\pi - \rho_{\pi_E}) - \lambda H(\pi) = D_{JS}(\rho_\pi, \rho_{\pi_E}) - \lambda H(\pi) \tag{1}$$

with φ_{GA} the cost regularizer function defined as:

$$\varphi_{GA}(c) = \begin{cases} \mathbb{E}_{\pi_E}[g(c(s,a))] & \text{if } c < 0 \\ +\infty & \text{otherwise} \end{cases}$$

where

$$g(x) = \begin{cases} -x - \log(1 - e^x) & \text{if } x < 0 \\ +\infty & \text{otherwise} \end{cases}$$

Equation 1 draws a connection between imitation learning and generative adversarial networks.

Key catch of the model

$\varphi_{GA}(\rho_\pi - \rho_{\pi_E})$ is a measure of discrepancy between the distribution of action occupancy generated by policy π and the distribution of action occupancy of expert policy π_E.

The cost function c is defined as: $c(s, a) = log(D(s, a))$.

$D_{JS}(\rho_\pi, \rho_{\pi_E})$ is the Jensen-Shannon divergence (a measure of distance or discrepancy between two probability distributions).
$\lambda \geq 0$ is the regularization constant for the policy.

$C : \mathbb{R}^{S \times A} = \{c : S \times A \to \mathbb{R}\}$ is the set of cost functions, where A is the set of all the actions and S is the set of all the states.

Let Π be the set of all stationary stochastic policies that select actions from A given a state $s \in S$.

For each policy $\pi \in \Pi$, we can define its occupancy measure $\rho_\pi : S \times A \to \mathbb{R}$ (i.e. the probability that a specific state-action pair is visited by a policy π) as $\rho_\pi(s, a) = \pi(a|s) \sum_{t=0}^{\infty} \gamma^t P(s_t = s|\pi)$, $\gamma \in (0, 1)$.

$H(\pi) = \mathbb{E}_\pi[-log(\pi(a|s))]$ is the e γ-discounted causal entropy (i.e. a measure of uncertainty in a discounted sequence with $\gamma \in (0, 1)$) of the policy π.

Key catch of the model

Algorithm 1 GAIL

Require: $\tau_E \sim \pi_E$ (expert trajectories), θ_0 (initial policy parameter), w_0 (initial discriminator parameter), π_θ (policy parametraized with weight θ), D_w (discrimanator network parametrized with weight w)

1: **for** $i = 0, 1, 2, \ldots$ **do**

2: Sample trajectories $\tau_i \sim \pi_{\theta_i}$, where π_{θ_i} is the probability distribution generated by the generator.

3: Update the discriminator parameters from w_i to w_{i+1} by solving Equation 1, finding a saddle point (π, D_w) of the expression

$$\mathbb{E}_\pi[log(D_w(s,a))] + \mathbb{E}_{\pi_E}[log(1 - D_w(s,a))] - \lambda H(\pi_\theta)$$

by using the gradient $\mathbb{E}_{\tau_i}[\nabla_w log(D_w(s,a)) + \mathbb{E}_{\tau_E}[\nabla_w log(1 - D_w(s,a))]]$

4: Take a policy step from θ_i to θ_{i+1} using the TRPO (Trust Region Policy Optimization) rule with cost function $c = log(D_{w_{i+1}}(s,a))$. Specifically, take a KL-constrained natural gradient step with

$$\mathbb{E}_{\tau_i}[\nabla_\theta log(\pi_\theta(a|s))Q(s,a) + \lambda \nabla_\theta H(\pi_\theta)$$

where $Q(s,a) = \mathbb{E}_{\tau_i}[log(D_{w+1}(s,a))|s_0 = s, a_0 = a]$

5: **end for**

Results

GAIL was compared to baseline algorithms, including behavioral cloning, feature expectation matching (FEM), game-theoretic apprenticeship learning (GTAL), and human experts, on nine physics-based control tasks.

On the classic control tasks, GAIL consistently outperformed behavioral cloning, FEM, and GTAL. However, behavioral cloning showed strong performance on the Reacher task, outperforming GAIL in terms of sample efficiency.

However, behavioral cloning was unable to achieve more than 60% performance on the Humanoid task, whereas GAIL achieved exact expert performance across all tested dataset sizes.

In other MuJoCo (Multi-Joint dynamics with Contact) environments, GAIL achieved a minimum of 70% expert performance across all tested dataset sizes and reached exact expert performance with larger datasets.

GAIL surpasses traditional imitation methods by offering a more flexible approach that does not rely on expert annotations.

The article demonstrates the effectiveness of GAIL in various imitation learning scenarios and highlighting its successful application in complex environments.

Comment

In conclusion, GAIL introducing a novel approach to imitation learning by exploiting generative adversarial techniques.

It demonstrates the ability to capture complex behaviors and accommodate diverse policy representations, leading to good sample efficiency and near-expert performance with smaller datasets.

However, achieving optimal results with GAIL requires careful selection of hyperparameters.

Overall, GAIL offers a flexible and effective approach to imitation learning, opening the way for further advancements in this field.