Imitation Learning On Atari Games

Usman Anwar

Information Technology University, Lahore.

June 9, 2020

Outline

Background

Generative Adversarial Imitation Learning (GAIL)

Metadata

Group Name	Imitation Learning On Atari Games
Group ID	G1W
Group Members	Usman Anwar
Paper Title	Generative Adversarial Imitation Learning
Paper Authors	Jonathan Ho (OpenAI) Stefano Ermon (Stanford)

Background

Generative Adversarial Imitation Learning (GAIL)

Reinforcement Learning: Introduction

In reinforcement learning, the objective is to learn a policy π which is a mapping from states $s \in \mathcal{S}$ to actions $a \in \mathcal{A}$. With each state-action pair, we associate a cost $c(s, a) : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$. The objective in reinforcement learning is to learn a policy π which minimizes the average cost an agent will incur over an episode.

$$\underset{\pi}{\operatorname{argmin}} \sum_{t=0}^{T} [c(s_t, a_t)] \tag{1}$$

What To Learn?

Reinforcement learning, and broadly machine learning, addresses the problem of learning from data. Generally, we decompose this problem of learning from data into two distinct phases:

- What to learn? Generally, human expert designs a cost function which informs the agent what task to perform and what kind of behaviour is desirable and should be learned.
- Learning From Data Given some cost function, an optimization scheme is generally developed which uses data and the specified cost function to actually learn a policy to do the specified task,

Is it always possible to specify a cost function?

Is it always possible to specify a cost function?

What is the cost function for driving a car?

What is the cost function for playing tennis?

What is the cost function for cooking?

Is it possible then to learn without specification of a cost function?

Inverse Reinforcement Learning (IRL)

Imitation Learning In imitation learning, the goal is to learn to imitate expert's policy given only finite only demonstrations performed by expert. Demonstrations are basically samples from experts policy and will be denoted as $\tau \sim \pi_E$.

Apprenticeship Learning In apprenticeship learning, the goal is to recover a reward function from the samples of rollouts by expert and then use this reward function to learn a policy which performs at least as well as the expert.

Problem Is Ill Posed

If we follow the principle of maximizing the log likelihood of the expert demonstrations, we arrive at an objective function of the following form

$$\operatorname*{argmax}_{c} \left(\operatorname*{argmin}_{\pi} \mathbb{E}_{\pi}[c(s,a)] \right) - \mathbb{E}_{\pi_E}[c(s,a)] \tag{2}$$

However, this objective function does not has a unique critical point.

Regularizer

We can use prior knowledge to inform a choice of regularizer which takes away at least some of the ambiguity present in the objective function above.

$$\operatorname*{argmax}_{c} -\psi(c) + \left(\operatorname*{argmin}_{\pi} \mathbb{E}_{\pi}[c(s,a)] \right) - \mathbb{E}_{\pi_E}[c(s,a)] \tag{3}$$

Maximum Entropy Inverse Reinforcement Learning

However, despite a regularizer, we can not eliminate the full ambiguity. [1] resolved this issue by prposing to keep the ambiguity in full by learning a cost function such that the *stochastic* policy learned by such a cost function has the maximum entropy.

$$\underset{c}{\operatorname{argmax}} \; -\psi(c) + \left(\underset{\pi}{\operatorname{argmin}} -H(\pi) + \mathbb{E}_{\pi}[c(s,a)] \right) - E_{\pi_E}[c(s,a)] \tag{4}$$

[1] B. D. Ziebart et al. "Maximum entropy inverse reinforcement learning.". In: *AAAI*. vol. 8. Chicago, IL, USA. 2008, pp. 1433–1438

Issues With This Approach

- This requires solving a forward RL problem inside the inner loop.

$$\underset{c}{\text{argmax}}\ -\psi(c) + \left(\underset{\pi}{\text{argmin}} -H(\pi) + \mathbb{E}_{\pi}[c(s,a)] \right) - E_{\pi_E}[c(s,a)] \quad (5)$$

  - This is REALLY expensive, which makes this approach hard to scale.
- Regularization is performed by pre-defining a set of functions $\mathcal{C}$ to which c must belong. A popular approach has been to assume c to lie in span of some pre-defined basis functions. Unfortunately, if c does not lie in this class, it can not be found by solving the above problem.

Background

Generative Adversarial Imitation Learning (GAIL)

GAIL [2]

Occupancy Measure Of A Policy

For a given policy π, we can define its occupancy measure $\rho_\pi : \mathcal{S} \times \mathcal{A} \to \mathbb{R}$ as $\rho_\pi(s, a) = \pi(a|s) \sum_{t=0}^{T} (P(s_t = s|\pi)$.

Intuitively, occupancy measure of a state-action pair is the unnormalized probability of a policy landing in that state s and then taking action a from that state.

[2] J. Ho and S. Ermon. "Generative adversarial imitation learning". In: *Advances in neural information processing systems*. 2016, pp. 4565–4573

GAIL

Instead of viewing the problem as finding a cost function which induces a policy with expected cost equal to expert policy, they view the problem as finding a cost function which induces a policy which has the same occupancy measure as the expert's policy. This has two benefits:

- Mapping from policy to occupancy measure is a bijection. So, there is a unqieu solution that could be found.
- This turns the problem of finding a policy into a probabiliy distribution estimation problem.
- This allows us to reduce the cost of expensive inner loop.

GAIL

Theorem

The optimal policy learnt under the cost function learnt by solving the optimization program

$$\underset{c}{\operatorname{argmax}} \; -\psi(c) + \left(\underset{\pi}{\operatorname{argmin}} -H(\pi) + \mathbb{E}_{\pi}[c(s,a)] \right) - E_{\pi_E}[c(s,a)] \tag{6}$$

is same as the solution to the following optimization problem where ψ^* is the frenchel conjugate of the regualarizer ψ.

$$\underset{\pi}{\operatorname{argmin}} -H(\pi) + \psi^*(\rho_\pi - \rho_{\pi_E}) \tag{7}$$

GAIL

Finding ψ

$$\underset{\pi}{\operatorname{argmin}} -H(\pi) + \psi^*(\rho_\pi - \rho_{\pi_E}) \tag{8}$$

We note that ρ_π and ρ_{π_E} are measures (probability distributions). So, one good choice for ψ^* will be a function which minimizes Jensen Shannon Divergence between them.

GAIL

Finding ψ

$$\underset{\pi}{\operatorname{argmin}} -H(\pi) + \psi^*(\rho_\pi - \rho_{\pi_E}) \tag{8}$$

We note that ρ_π and ρ_{π_E} are measures (probability distributions). So, one good choice for ψ^* will be a function which minimizes Jensen Shannon Divergence between them. Following choice of * does that

$$\psi^*(\rho_\pi - \rho_{\pi_E}) = \sup_{D \in (0,1)^{\mathcal{S}\times\mathcal{A}}} \mathbb{E}_\pi\left[\log(D(s,a)\right] + \mathbb{E}_{\pi_E}[\log(1 - D(s,a))] \tag{9}$$

GAIL

Finding ψ

$$\underset{\pi}{\operatorname{argmin}} -H(\pi) + \psi^*(\rho_\pi - \rho_{\pi_E}) \tag{8}$$

We note that ρ_π and ρ_{π_E} are measures (probability distributions). So, one good choice for ψ^* will be a function which minimizes Jensen Shannon Divergence between them. Following choice of * does that

$$\psi^*(\rho_\pi - \rho_{\pi_E}) = \sup_{D \in (0,1)^{\mathcal{S}\times\mathcal{A}}} \mathbb{E}_\pi \left[\log(D(s,a)\right] + \mathbb{E}_{\pi_E}[\log(1 - D(s,a))] \tag{9}$$

The following choice of ψ has the frenchel conjugate of the above form.

$$\psi(c) \doteq \begin{cases} \mathbb{E}_{\pi_E}[-c(s,a) - \log(1 - e^{c(s,a)})], & \text{if} \quad c(s,a) < 0 \\ +\infty & \text{otherwise.} \end{cases}$$

GAIL

Finding ψ

$$\psi(c) \doteq \begin{cases} \mathbb{E}_{\pi_E}[-c(s,a) - \log(1 - e^{c(s,a)})], & \text{if} \quad c(s,a) < 0 \\ +\infty & \text{otherwise.} \end{cases}$$

This regularizer has couple of nice properties:

- It admits all cost functions which are negative everywhere. This class is sufficiently expressive for almost all cases.
- If the expert demonstrations are assigned high cost then this regularize penalizes c heavily.
- This is average over expert demonstrations, so, it can automatically adjust to new demonstrations and datasets and does not need to be tuned for any problem.

GAIL

Final Objective

GAIL attempts to find a saddle point (π, D) of the following expression.

$$\mathbb{E}_{\pi}[\log(D(s,a))] + \mathbb{E}_{\pi_E}[\log(1 - D(s,a))] - \lambda H(\pi) \tag{10}$$

where π and D are both represented as neural networks with weights θ and w respectively. D is a discriminator network and λ is a hyperparameter which can be used to control the role of entropy maximization.

GAIL

Algorithm

for $i = 1, 2, 3, \ldots$:

1. Sample trajectories from current policy $\tau_i \sim \pi_{\theta_i}$.
2. Update discriminator parameters from w_i to w_{i+1} with the gradient

$$\hat{\mathbb{E}}_{\tau_i}[\nabla_w \log(D(s,a))] + \hat{\mathbb{E}}_{\tau_E}[\nabla_w \log(1 - D(s,a))] \tag{11}$$

3. Update policy network parameters from θ_i to θ_{i+1} using Trust Region Policy Update (TRPO) update rule with cost function $\log(D_{w_{i+1}}(s,a))$ and gradient

$$\hat{\mathbb{E}}_{\tau_i}[\nabla_\theta \log \pi_\theta(a|s) \log(D_{w_{i+1}}(s,a))] - \lambda \nabla_\theta H(\pi_\theta) \tag{12}$$

GAIL

Comparison With GAN

The approach has quite a few parallels with Generative Adversarial Network (GAN) [3]. The key distinction, however, is that sampling process in GAN is differentiable, which allows us to train GAN in an end to end fashion.

However, in GAIL, the sampling involves a non-differentiable environment/simulator, so, we need to run a spearate policy search algorithm (i.e. TRPO) to update parameters of policy network. This still amounts to doing forward RL but key difference here, compared to other works, is that forward RL loop need not be run till convergence.

[3] I. Goodfellow et al. "Generative adversarial nets". In: *Advances in neural information processing systems*. 2014, pp. 2672–2680

GAIL

Results

GAIL was tested on 9 physics based control tasks from OpenAI Gym environment. These tasks incluced Humanoid (make a human like robot walk), Ant (make a 3D four legged robot walk), Reacher (make a 2D arm like robot reach a specific location), Hopper (make a 2D robot hop). On all tasks, GAIL was able to match expert's performance.

GAIL

Discussion

- This is considered one of seminal works in IRL and has inspired a range of derivative works.
  - InfoGAIL - GAIL analogue of infoGAN.
  - Multi Agent GAIL - GAIL for multi agent environments like playing a cricket match
  - Differentiable GAIL - Make sampling process differentiable using a model.
- Still the de-facto baseline against which works in IRL compare themselves.

GAIL On Atari

Challenges

There are two major challenges in this:

- Original GAIL was used on tasks with fully observable state. In Atari the state space is partially observable which is a big challenge.
- The state space in Atari is very high dimensional (thousand of pixels) and original GAIL was used on tasks with comparitively small dimensions. For example, in humanoid full state dimension is only 47.

Thank You.