Adversarial Imitation Learning via Random Search

Authors: MyungJae Shin (Presenter) and Joongheon Kim
School of Computer Science and Engineering, Chung-Ang University, Seoul, Republic of Korea
Emails: mjshin.cau@gmail.com, joongheon@gmail.com
Sites: github.com/170928, https://sites.google.com/site/joongheonkim/
Reinforcement Learning

**Goal**: Learn policies
High-dimensional & raw observations

![Diagram of reinforcement learning process](image-url)
**Input**: expert behavior generated by expert $\pi_E$

$$\{(s^i_0, a^i_0, s^i_1, a^i_1, \ldots)\}_{i=1}^N \sim \pi_E$$

**Goal**: learn cost function or policy
Imitation Learning

Environment Model (MDP)

Reward Function $R$

Reinforcement Learning (RL)

Optimal Policy $\pi$

$$RL(R) = \arg \min_{\pi} \mathbb{E}_{\pi} [R(s, a)] - H(\pi)$$
Imitation Learning

\[ \max_R \left( \min_{\pi} \mathbb{E}_{\pi} [R(s, a)] - H(\pi) \right) - \mathbb{E}_{\pi_E} [R(s, a)] \]
Imitation Learning

\[
\max_R -\psi(R) + \left( \min_\pi \mathbb{E}_\pi [R(s, a)] - H(\pi) \right) - \mathbb{E}_{\pi_E} [R(s, a)]
\]

\[\min_\pi \psi^* (\rho_\pi - \rho_{E\pi}) - H(\pi)\]

\(\rho_\pi = \text{occupancy measure}\)

\(\Rightarrow\) Distribution of state-action pairs encountered when navigating the environment with the policy

\(\rho_{E\pi} = \text{expert's occupancy measure}\)
Imitation Learning

[Theorem]

ψ regularized inverse reinforcement learning implicitly, seeks a policy whose occupancy measure is close to the expert’s, as measured by ψ*

- Typical IRL finds a cost function such that the expert policy is uniquely optimal
- IRL as a procedure that tries to induce a policy that matches the expert’s occupancy measure (generative model)

Generative Adversarial Imitation Learning (GAIL), NIPS 2016

Use this regularizer

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

Generative Adversarial Networks, Ian J. Goodfellow, NIPS 2014
Adversarial Imitation Learning via Random Search

Generative Adversarial Imitation Learning (GAIL), *NIPS 2016*
Based on the output of the discriminator (Generative Adversarial Networks, Ian J. Goodfellow, 2014), we could know the difference between the distribution of expert data and that of agent.
Adversarial Imitation Learning via Random Search

\[ \text{minimize } \mathbb{E}_\pi[\log(D(s, a))] + \mathbb{E}_{\pi_E}[\log(1 - D(s, a))] \]

**$D(s, a)$**: Probability between 0 and 1

The probability that the input data sample is the expert data sample
Challenge
A lot of interaction with the environment is required to optimize the policy through GAIL framework.
Adversarial Imitation Learning via Random Search

Usually in AI:

\[ f'(x) = \frac{df}{dx} \]

Proposed method

\[ f'(a) = \frac{f(a + h) - f(a)}{h} \]
Adversarial Imitation Learning via Random Search

Policy
\[ \theta_t \]

\[
\begin{array}{ccc}
  w_{1,1} & w_{1,2} & w_{1,3} \\
  w_{2,1} & w_{2,2} & w_{2,3}
\end{array}
\]

\[ \delta_i \]

\[ \delta = \{ \delta_1, \delta_2, \ldots, \delta_N \} \]

Gaussian Noise

\[
\begin{array}{ccc}
  \begin{array}{ccc}
    0.14 & 0.37 & 0.49 \\
    0.96 & 0.25 & 0.94
  \end{array} & \begin{array}{ccc}
    w_{1,1} & + & 0.14 \\
    w_{2,1} & + & 0.96
  \end{array} & \begin{array}{ccc}
    w_{1,2} & + & 0.37 \\
    w_{2,2} & + & 0.25
  \end{array} & \begin{array}{ccc}
    w_{1,3} & + & 0.49 \\
    w_{2,3} & + & 0.94
  \end{array}
\end{array}
\]

\[
\begin{array}{ccc}
  \begin{array}{ccc}
    -0.14 & -0.37 & -0.49 \\
    -0.96 & -0.25 & -0.94
  \end{array} & \begin{array}{ccc}
    w_{1,1} & - & 0.14 \\
    w_{2,1} & - & 0.96
  \end{array} & \begin{array}{ccc}
    w_{1,2} & - & 0.37 \\
    w_{2,2} & - & 0.25
  \end{array} & \begin{array}{ccc}
    w_{1,3} & - & 0.49 \\
    w_{2,3} & - & 0.94
  \end{array}
\end{array}
\]
Adversarial Imitation Learning via Random Search

\[ \mathbf{R}_{d-pos} \]

\[
\begin{align*}
\mathbf{w}_{1,1} &+ d_{1,1} \\
\mathbf{w}_{1,2} &+ d_{1,2} \\
\mathbf{w}_{1,3} &+ d_{1,3} \\
\mathbf{w}_{2,1} &+ d_{2,1} \\
\end{align*}
\]

\[ \mathbf{R}_{e-pos} \]

\[
\begin{align*}
\mathbf{w}_{1,1} &+ e_{1,1} \\
\mathbf{w}_{1,2} &+ e_{1,2} \\
\mathbf{w}_{1,3} &+ e_{1,3} \\
\mathbf{w}_{2,1} &+ e_{2,1} \\
\end{align*}
\]

\[ \mathbf{R}_{f-pos} \]

\[
\begin{align*}
\mathbf{w}_{1,1} &+ f_{1,1} \\
\mathbf{w}_{1,2} &+ f_{1,2} \\
\mathbf{w}_{1,3} &+ f_{1,3} \\
\mathbf{w}_{2,1} &+ f_{2,1} \\
\end{align*}
\]

\[ \mathbf{R}_{g-pos} \]

\[
\begin{align*}
\mathbf{w}_{1,1} &+ g_{1,1} \\
\mathbf{w}_{1,2} &+ g_{1,2} \\
\mathbf{w}_{1,3} &+ g_{1,3} \\
\mathbf{w}_{2,1} &+ g_{2,1} \\
\end{align*}
\]

\[ \mathbf{R}_{d-neg} \]

\[
\begin{align*}
\mathbf{w}_{1,1} &- d_{1,1} \\
\mathbf{w}_{1,2} &- d_{1,2} \\
\mathbf{w}_{1,3} &- d_{1,3} \\
\mathbf{w}_{2,1} &- d_{2,1} \\
\end{align*}
\]

\[ \mathbf{R}_{e-neg} \]

\[
\begin{align*}
\mathbf{w}_{1,1} &- e_{1,1} \\
\mathbf{w}_{1,2} &- e_{1,2} \\
\mathbf{w}_{1,3} &- e_{1,3} \\
\mathbf{w}_{2,1} &- e_{2,1} \\
\end{align*}
\]

\[ \mathbf{R}_{f-neg} \]

\[
\begin{align*}
\mathbf{w}_{1,1} &- f_{1,1} \\
\mathbf{w}_{1,2} &- f_{1,2} \\
\mathbf{w}_{1,3} &- f_{1,3} \\
\mathbf{w}_{2,1} &- f_{2,1} \\
\end{align*}
\]

\[ \mathbf{R}_{g-neg} \]

\[
\begin{align*}
\mathbf{w}_{1,1} &- g_{1,1} \\
\mathbf{w}_{1,2} &- g_{1,2} \\
\mathbf{w}_{1,3} &- g_{1,3} \\
\mathbf{w}_{2,1} &- g_{2,1} \\
\end{align*}
\]

\[ \mathbf{R}_{d} - \mathbf{p} o s \]

\[ \mathbf{R}_{e} - \mathbf{n} e g \]

\[ \mathbf{R}_{f} - \mathbf{p} o s \]

\[ \mathbf{R}_{g} - \mathbf{n} e g \]
Adversarial Imitation Learning via Random Search

\[ \text{Policy } \theta_{t+1} = \text{Policy } \theta_t + \begin{bmatrix} R_{d-pos} - R_{d-neg} \end{bmatrix} * \begin{bmatrix} d_{11} & d_{12} & d_{13} \\ d_{21} & d_{22} & d_{23} \end{bmatrix} + \begin{bmatrix} R_{e-pos} - R_{e-neg} \end{bmatrix} * \begin{bmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \end{bmatrix} \]
Adversarial Imitation Learning via Random Search

\[
\begin{align*}
    w_{1,1} &+ 0.14 \\
    w_{1,2} &+ 0.37 \\
    w_{1,3} &+ 0.49 \\
    w_{2,1} &+ 0.96 \\
    w_{2,2} &+ 0.25 \\
    w_{2,3} &+ 0.94 \\
\end{align*}
\]

\[
\begin{align*}
    w_{1,1} &- 0.14 \\
    w_{1,2} &- 0.37 \\
    w_{1,3} &- 0.49 \\
    w_{2,1} &- 0.96 \\
    w_{2,2} &- 0.25 \\
    w_{2,3} &- 0.94 \\
\end{align*}
\]

Reward

\[ r(\theta_t - \nu \delta_i) \]

Expert Demonstration

\[ r(\theta_t + \nu \delta_i) \]

Sample Trajectories
**Adversarial Imitation Learning via Random Search**

**Reward**
- \( r(\theta_t - \nu \delta_i) \)
- \( r(\theta_t + \nu \delta_i) \)

**Policy Update**
\[
\theta_{t+1} = \theta_t + \frac{\alpha}{\sigma_R} \sum_{i=1}^{N} (r(\theta_t + \nu \delta_i) - r(\theta_t - \nu \delta_i)) \delta_i
\]
Adversarial Imitation Learning via Random Search
Adversarial Imitation Learning via Random Search

Scaled Rewards vs. Number of expert trajectories in the dataset

- AILSRS
- BC
- GAIL
- Expert Rewards
Thank You