CS 277: Control and Reinforcement Learning

Winter 2024

Lecture 14: Inverse RL

Roy Fox
Department of Computer Science
School of Information and Computer Sciences
University of California, Irvine

Logistics

assignments

- Quiz 7 due next Wednesday
- Exercise 4 + Quiz 8 will be due Week 10
- Exercise 5 will be due Week 11

Today's lecture

Sparse rewards

IRL

MaxEnt IRL

GAIL

Relation between RL and IL

- What makes RL harder than IL?
  - IL: teacher policy $\pi_e(a \mid s)$ indicates a good action to take in s
  - RL: $r(s, a)$ does not indicate a globally good action; $Q^*(s, a)$ does, but it's nonlocal
- But didn't we see an equivalence between RL and IL?
  - NLL loss in BC: $\nabla_\theta \mathbb{E}[\log \pi_\theta(a \mid s)]$
    - s and a sampled from teacher distribution
  - PG loss: $\nabla_\theta \mathbb{E}[\log \pi_\theta(a \mid s) R]$
    - s and a sampled from learner distribution

Informational quantities: refresher

- Entropy: $\mathbb{H}[p(a)] = -\mathbb{E}_{a\sim p}[\log p(a)] = -\sum_a p(a)\log p(a)$
- Conditional entropy: $\mathbb{H}[\pi|s] = -\mathbb{E}_{a\sim\pi}[\log \pi(a|s)]$
- Expected conditional entropy: $\mathbb{H}[\pi] = \mathbb{E}_{s\sim p_\pi}[\mathbb{H}[\pi|s]] = -\mathbb{E}_{s,a\sim p_\pi}[\log \pi(a|s)]$
- Expected relative entropy: $\mathbb{D}[\pi\|\pi'] = \mathbb{E}_{s,a\sim p_\pi}\left[\log \frac{\pi(a|s)}{\pi'(a|s)}\right]$
- Expected cross entropy (aka NLL): $-\mathbb{E}_{s,a\sim p_\pi}[\log \pi'(a|s)]$
  - $\mathbb{D}[\pi\|\pi'] = \text{NLL} - \mathbb{H}[\pi]$

IL as sparse-reward RL

- NLL BC: maximize $\mathbb{E}_{s,a\sim p_e}[\log \pi_\theta(a|s)] = -\mathbb{D}[\pi_e \| \pi_\theta] - \mathbb{H}[\pi_e]$

  constant in θ

  - Experience from teacher distribution p_e
    - RL: experience from learner distribution p_θ
  - "Return" $R = 1_{\text{success}}$ for successful trajectory
    - RL: $r_t = r(s_t, a_t)$ in every step
- Sparse reward = most rewards are 0 $\implies$ rare learning signal
  - $R = 1$ on success = very sparse; but doesn't IL provide dense learning signal?

IL as dense-reward RL

- What if instead we minimize the other relative entropy?

teacher labeling of learner states/actions
as in DAgger

$$\mathbb{D}[\pi_\theta \| \pi_e] = -\mathbb{E}_{s,a \sim p_\theta}[\log \pi_e(a \mid s)] - \mathbb{H}[\pi_\theta]$$

  - This is exactly the RL objective, with $r(s, a) = \log \pi_e(a \mid s)$ and entropy regularizer
  - Now $r(s, a)$ does give global information on optimal action
  - In fact, with deterministic teacher, $r(s, a) = -\infty$ for any suboptimal action
- The same return can be viewed as dense reward or sum of sparse rewards
  - Can we do the same in proper RL?

Reward shaping

- Ideal reward: $r(s, a) = -\infty$ for any suboptimal action $\implies$ as hard to provide as π^*
  - We need supervision signal that's sufficiently easy to program $\implies$ generate more data
- Sparse reward functions may be easier than dense ones
  - E.g., may be easy to identify good goal states, safety violations, etc.
- Reward shaping: art of adjusting the reward function for easier RL; some tips:
  - Reward "bottleneck states": subgoals that are likely to lead to bigger goals
  - Break down long sequences of coordinated actions $\implies$ better exploration
    - E.g. reward beacons on long narrow paths, for exploration to stumble upon

Today's lecture

Sparse rewards

IRL

MaxEnt IRL

GAIL

Learning rewards from demonstrations

- RL: rewards → policy; IL: demonstrations → policy
- Inverse Reinforcement Learning (IRL): demonstrations → reward function
    - Better understand agents (humans, animals, users, markets)
        - Preference elicitation, teleology (the “what for” of actions), theory of mind, language
    - First step toward Apprenticeship Learning: demos → rewards → policy
        - Infer the teacher's goals and learn to achieve them; overcome suboptimal demos
        - Partly model-based (learn r but not p); may be easier to learn, generalize, transfer
        - Teacher and learner can have different action spaces (e.g., human → robot)

Inverse Reinforcement Learning (IRL)

- Given a dataset of demonstration trajectories $\mathscr{D} = \{\xi_i\}$

  $r(s)$ **expressive enough**

- Find teacher's reward function $r : \mathscr{S} \times \cancel{\mathscr{A}} \to \mathbb{R}$
  - Principle: demonstrated actions should achieve high expected return
- IRL is ill-defined
  - How low is the reward for states and actions not in $\mathscr{D}$?
  - How is the reward distributed along the trajectory?
    - Sparse rewards = identify "subgoal" states; dense = score each step, as hard as IL
  - Demonstrator can be fallible = take suboptimal actions; how much?

Feature matching

- Assume linear reward $r_\theta(s) = \theta^\intercal f_s$ in given state features $f_s \in \mathbb{R}^d$

  $t \sim \mathrm{Geom}(1-\gamma)$

  missing const: $(1-\gamma)$

  - Value = $J_\theta^\pi = \sum_t \gamma^t \mathbb{E}_{s_t \sim p_\pi}[\theta^\intercal f_{s_t}] = \mathbb{E}_{s \sim p_\pi}[\theta^\intercal f_s]$, with $p_\pi(s) \propto \sum_t \gamma^t p_\pi(s_t)$

- Teacher optimality: expert value $J_\theta^{\pi^*}$ higher than any other policy's value J_θ^π
  - Find θ that maximizes the gap $J_\theta^{\pi^*} - J_\theta^\pi$; but for which π?
  - Apprenticeship Learning: find π that maximizes J_θ^π ; but for which θ?

- Solve: $\max_\theta \min_\pi \{J_\theta^{\pi^*} - J_\theta^\pi\} = \max_\theta \min_\pi \{\mathbb{E}_{s \sim p^*}[\theta^\intercal f_s] - \mathbb{E}_{s \sim p_\pi}[\theta^\intercal f_s]\}$
  - Approximate $s \sim p^*$ with $s \sim \mathscr{D}$

Feature matching

- Solving $\max_\theta \min_\pi \{\mathbb{E}_{s\sim p^*}[\theta^\intercal f_s] - \mathbb{E}_{s\sim p_\pi}[\theta^\intercal f_s]\}$

Algorithm Feature Matching

Initialize policy set $\Pi = \{\pi_0\}$

repeat

Solve Quadratic Program: $\max_{\eta, \|\theta\|_2 \le 1} \eta$ — θ must be bounded, or solution at ∞

s.t. $\mathbb{E}_{s\sim\mathcal{D}}[\theta^\intercal f_s] \geq \mathbb{E}_{s\sim p_\pi}[\theta^\intercal f_s] + \eta \quad \forall \pi \in \Pi$

$\pi \leftarrow$ optimal policy for $r_\theta(s) = \theta^\intercal f_s$

Add π to Π

- On convergence: π optimal for θ (no gap), while no θ increases the gap
  - $\Rightarrow \mathbb{E}_{s\sim\mathscr{D}}[\theta^\intercal f_s] \approx \mathbb{E}_{s\sim p_\pi}[\theta^\intercal f_s]$ for all $\theta \Rightarrow \mathbb{E}_{s\sim\mathscr{D}}[f_s] \approx \mathbb{E}_{s\sim p_\pi}[f_s]$ — feature matching

Today's lecture

Sparse rewards

IRL

MaxEnt IRL

GAIL

Modeling bounded teachers

- An expert teacher maximizes the value $J_\theta^{\pi^*} = \sum_t \gamma^t \mathbb{E}_{s_t \sim p^*}[\theta^\intercal f_{s_t}] = \mathbb{E}_{\xi \sim p^*}[\theta^\intercal f_\xi]$
  - With trajectory-summed features $f_\xi = \sum_t \gamma^t f_{s_t}$
- Assume teacher has unintentional / uninformed prior policy π_0
  - Bounded rationality: cost to intentionally diverge $\mathbb{D}[\pi^* \| \pi_0]$ (with π_0 uniform: $\mathbb{H}[\pi^*]$)
  - Total cost: $\sum_t \mathbb{E}_{(s_t,a_t) \sim p^*}\left[\log \frac{\pi^*(a_t \mid s_t)}{\pi_0(a_t \mid s_t)}\right] = \mathbb{E}_{\xi \sim p^*}\left[\log \frac{p^*(\xi)}{p_0(\xi)}\right] = \mathbb{D}[p^*(\xi) \| p_0(\xi)]$

for simplicity, assume $\tau = 1$

- Bounded optimality: $\max_{\pi^*} \mathbb{E}_{\xi \sim p^*}[\theta^\intercal f_\xi] - \tau \mathbb{D}[p^* \| p_0]$

Bounded optimality: naïve solution

- Bounded optimality: $\max_{\cancel{\pi^*}\ p^*} \mathbb{E}_{\xi\sim p^*}[\theta^\intercal f_\xi] - \mathbb{D}[p^*\|p_0]$

  $\mathbb{E}_{\xi\sim p^*}[\log p^*(\xi) - \log p_0(\xi)]$

  - Naïve solution: allow any distribution p^* over trajectories
  - No need to be consistent with dynamics $p(s'|s,a) \Rightarrow p^*$ may be unachievable
- Add the constraint $\sum_\xi p^*(\xi) = 1$ with Lagrange multiplier λ
- Differentiate by $p^*(\xi)$ and $= 0$ to optimize

$$\theta^\intercal f_\xi - \log p^*(\xi) + \log p_0(\xi) - 1 + \lambda = 0 \Longrightarrow p^*(\xi) = \frac{p_0(\xi)\exp(\theta^\intercal f_\xi)}{\sum_{\bar\xi} p_0(\bar\xi)\exp(\theta^\intercal f_{\bar\xi})}$$

IRL with bounded teacher

- Assume that demonstrations are distributed $p_\theta(\xi) = \frac{1}{Z_\theta} p_0(\xi) \exp(\theta^\intercal f_\xi)$
  - With partition function $Z_\theta = \mathbb{E}_{\bar{\xi} \sim p_0}[\exp(\theta^\intercal f_{\bar{\xi}})]$
- Find θ that minimizes NLL of demonstrations

$$\nabla_\theta \log p_\theta(\xi) = \nabla_\theta(\theta^\intercal f_\xi - \log Z_\theta) = f_\xi - \frac{1}{Z_\theta} \nabla_\theta Z_\theta$$

$$= f_\xi - \frac{1}{Z_\theta} \mathbb{E}_{\bar{\xi} \sim p_0}[\exp(\theta^\intercal f_{\bar{\xi}}) f_{\bar{\xi}}] = f_\xi - \mathbb{E}_{\bar{\xi} \sim p_\theta}[f_{\bar{\xi}}]$$

  - To compute gradient, we need p_θ, but how to compute Z_θ?

Computing Z_θ: backward recursion

- Partition function: $Z_\theta = \mathbb{E}_{\xi \sim p_0}[\exp(\theta^\intercal f_\xi)]$

- Compute Z_θ recursively backward: like a value function, but + becomes ·

$$Z_\theta(s_t, a_t) = \mathbb{E}_{p_0}[\exp(\theta^\intercal f_{\xi \geq t}) \,|\, s_t, a_t] = \exp(\theta^\intercal f_{s_t}) \mathbb{E}_{(s_{t+1}|s_t, a_t) \sim p}[Z_\theta(s_{t+1})]$$

$$Z_\theta(s_t) = \mathbb{E}_{p_0}[\exp(\theta^\intercal f_{\xi \geq t}) \,|\, s_t] = \mathbb{E}_{(a_t|s_t) \sim \pi_0}[Z_\theta(s_t, a_t)]$$

part of the normalizer involving trajectories following (s_t, a_t)

- How to get a policy from Z_θ?

everything up to s_t cancels out

  - Marginalize: $\pi_\theta(a_t \,|\, s_t) = \dfrac{p_\theta(\xi \,|\, s_t, a_t)}{p_\theta(\xi \,|\, s_t)} = \dfrac{p_0(\xi_{\geq t} \,|\, s_t, a_t)\exp(\theta^\perp f_{\xi \geq t}) \cdot Z_\theta(s_t)}{Z_\theta(s_t, a_t) \cdot p_0(\xi_{\geq t} \,|\, s_t)\exp(\theta^\perp f_{\xi \geq t})} = \pi_0(a_t \,|\, s_t)\dfrac{Z_\theta(s_t, a_t)}{Z_\theta(s_t)}$

consistent π may not even exist

  - This π_θ is not globally consistent $p_\theta(\xi) \neq p_{\pi_\theta}(\xi)$, $p_\theta(\xi)$ ignores the dynamics

MaxEnt IRL

- For each sample $\xi \sim \mathscr{D}$:
  - Compute $Z_\theta = \mathbb{E}_{\xi \sim p_0}[\exp(\theta^\intercal f_\xi)]$ recursively backward
  - Compute $\mathbb{E}_{\bar{\xi} \sim p_{\pi_\theta}}[f_{\bar{\xi}}]$ recursively forward
  - Take a gradient step to improve θ: $\nabla_\theta \log p_\theta(\xi) \approx f_\xi - \mathbb{E}_{\bar{\xi} \sim p_{\pi_\theta}}[f_{\bar{\xi}}]$
- At the optimum: feature matching $\mathbb{E}_{\xi \sim \mathscr{D}}[f_\xi] = \mathbb{E}_{\xi \sim p_{\pi_\theta}}[f_\xi]$
  - MaxEnt IRL approximates $\max_\theta \mathbb{H}[\pi_\theta]$ s.t. $\mathbb{E}_{\xi \sim \mathscr{D}}[f_\xi] = \mathbb{E}_{\xi \sim p_{\pi_\theta}}[f_\xi]$

Limitations:

- Requires dynamics p
- Assumes $p_\theta = p_{\pi_\theta}$
- Assumes $\mathscr{D} = p^*$

Today's lecture

Sparse rewards

IRL

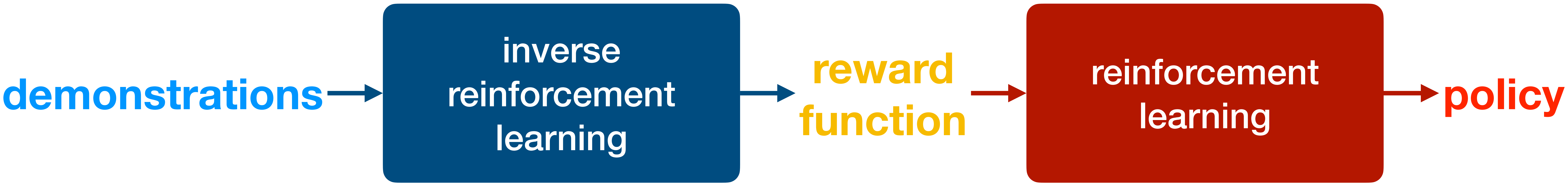
MaxEnt IRL

GAIL

IRL: downstream tasks

- One IRL motivation: learn reward function for downstream tasks

...such as RL



- IL = RL ∘ IRL (composition of RL on IRL)

- Our algorithms already learn π as part of learning θ for $r : s \mapsto \theta^\intercal f_s$
  - Let's directly optimize IRL for the overall IL task = learn good π

IL as RL ∘ IRL

- Entropy-regularized RL: $\max_{\pi \in \Pi} \left\{ \mathbb{E}_{s \sim p_\pi}[r(s)] + \mathbb{H}[\pi] \right\}$

regularization over reward function space

- MaxEnt IRL: $\max_{r \in \mathbb{R}^{\mathcal{S}}} \left\{ \mathbb{E}_{s \sim p_e}[r(s)] - \max_{\pi \in \Pi} \left\{ \mathbb{E}_{s \sim p_\pi}[r(s)] + \mathbb{H}[\pi] \right\} \right\} - \psi(r)$

- For any π, our objective with respect to r is:

$$\psi^*(p_e - p_\pi) = \max_{r \in \mathbb{R}^{\mathcal{S}}} \left\{ \overbrace{(p_e - p_\pi)}^{\in \mathbb{R}^{\mathcal{S}}} \cdot r - \psi(r) \right\}$$

  - This form of function $\psi^* : \mathbb{R}^{\mathcal{S}} \to \mathbb{R}$ is called the convex conjugate of ψ

Reward-function regularizers

$$\psi^*(p_e - p_\pi) = \max_{r \in \mathbb{R}^{\mathcal{S}}} \left\{ (p_e - p_\pi) \cdot r - \psi(r) \right\}$$

- Without regularizer: $\psi = 0 \implies$ solution only exists when $p_e = p_\pi$
  - $\implies$ learner achieves teacher's state distribution: perfect solution, but hard to find
- Hard linearity constraint: $\psi(r) = \begin{cases} 0 & \text{if } r(s) = \theta^\intercal f_s \\ \infty & \text{otherwise} \end{cases}$
  - $\implies$ max-entropy feature matching (MaxEnt IRL)
  - Great when the reward function really is linear in f_s, otherwise no guarantees

Generative Adversarial Networks (GANs)

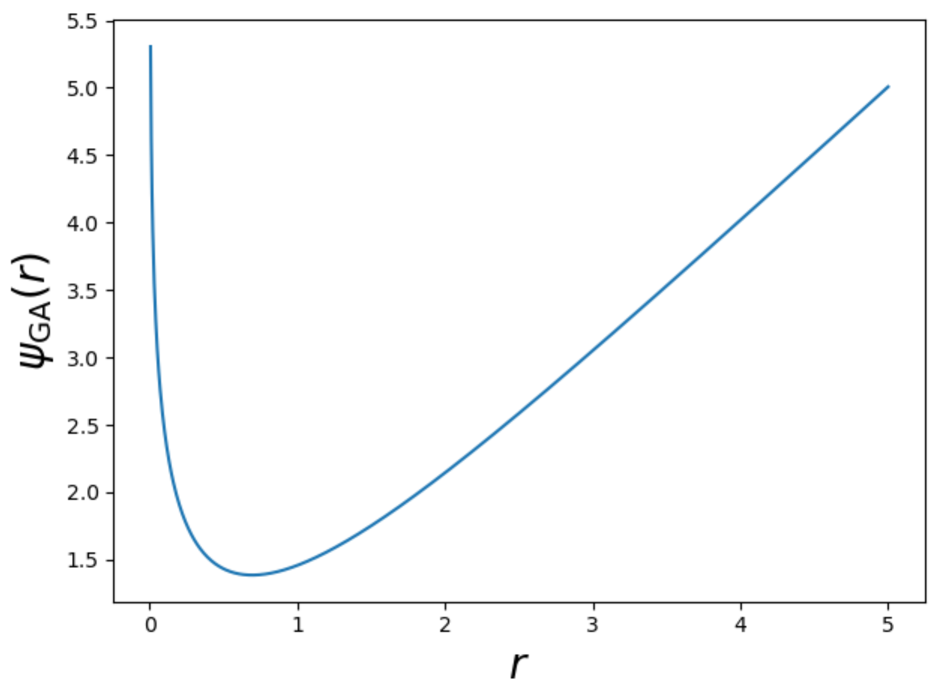
- Train generative model $p_\theta(s)$ to generate states / observations
  - Can we focus the training on failure modes?

- Also train discriminator $D_\phi(s) \in [0,1]$ to score instances
  - Kind of like a critic: are generated instances good?
- $D_\phi(s)$ predicts the probability $p(s \text{ generated by learner} \mid s) = \dfrac{p_\theta(s)}{p_\theta(s) + p_e(s)}$
  - Trained with cross-entropy loss: $\max_\phi \left\{ \mathbb{E}_{s \sim p_\theta}[\log D_\phi(s)] + \mathbb{E}_{s \sim p_e}[\log(1 - D_\phi(s))] \right\}$
- The generator tries to fool the discriminator: $\min_\theta \mathbb{E}_{s \sim p_\theta}[\log D_\phi(s)]$

Teacher-based reward-function regularizer

- Consider the regularizer

$$\psi_{\text{GA}}(r) = \mathbb{E}_{s\sim p_e}[r(s) - \log(1 - \underbrace{\exp(-r(s))}_{D(s)})]$$



- It's convex conjugate is:

$$\begin{aligned}\psi^*_{\text{GA}}(p_e - p_\pi) &= \max_{r\in\mathbb{R}^{\mathcal{S}}} \left\{(p_e - p_\pi)\cdot r - \psi(r)\right\} \\ &= \max_{r\in\mathbb{R}^{\mathcal{S}}}[r(s) - r(s) + \log(1 - D(s))] - \mathbb{E}_{s\sim p_\pi}[\overbrace{r(s)}^{-\log D(s)}] \\ &= \mathbb{E}_{s\sim p_\pi}[\log D(s)] + \mathbb{E}_{s\sim p_e}[\log(1 - D(s))]\end{aligned}$$

  - $\Longrightarrow$ GAN: generator p_π imitating teacher p_e; discriminator $D(s) = \exp(-r(s))$

Generative Adversarial Imitation Learning (GAIL)

Input: demonstration dataset $\mathcal{D}_T \sim p_T$

repeat

$\quad \mathcal{D}_L \leftarrow$ roll out π_θ

$\quad$ take discriminator gradient ascent step

$$\mathbb{E}_{s\sim\mathcal{D}_L}[\nabla_\phi \log D_\phi(s)] + \mathbb{E}_{s\sim\mathcal{D}_T}[\nabla_\phi \log(1 - D_\phi(s))]$$

$\quad$ take entropy-regularized policy gradient step with reward $r(s) = -\log D_\phi(s)$

- We've already seen one entropy-regularized PG algorithm: TRPO
  - More next time

Recap

- To understand behavior: infer the intentions of observed agents
- If teacher is optimal for a reward function
  - The reward function should make an optimizer imitate the teacher
  - State (or state–action) distribution of learner should match the teacher
- In this view, Inverse Reinforcement Learning (IRL) is a game:
  - Reward is optimized to show how much the teacher is better than the learner
  - Learner optimizes for the reward
  - Reward is like a discriminator (high = probably teacher); learner like a generator