UCI University of California, Irvine

CS 295: Optimal Control and Reinforcement Learning

Winter 2020

Lecture 14: Inverse Reinforcement Learning

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

Today's lecture

- Inverse Imitation Learning (IRL):
  - learning a reward function from demonstrations
- Feature matching
- Maximum Entropy IRL
  - Feature matching with entropy regularization
- GAIL

Learning rewards from demonstrations

- If we have demonstrations, why learn rewards?
  - Preference elicitation: better understand humans, animals, users, markets
  - Imitation learning: transfer between action spaces (human → robot)
  - Reinforcement learning: optimize for the intention of fallible teachers
  - Rewards may be easier to model, generalize, transfer
  - Teleology ("what'd you do that for"), theory of mind, are part of natural language

Inverse Reinforcement Learning (IRL)

- Given a dataset of demonstration trajectories $\mathcal{D} = \{\xi_i\}_i$
- Find the demonstrator's reward function $r_\theta : \mathcal{S} \to \mathbb{R}$
- The result is underdetermined
  - If we only see positive examples, no telling how inclusive the reward should be
  - How dense should the reward be (perhaps not uniformly?)
    - Learning very dense rewards may not give benefits over IL
  - Teacher can be fallible

Feature matching

- Suppose we have an extractor of relevant state features $f_s \in \mathbb{R}^d$
- Assume a linear reward: $r_\theta(s) = \theta^\intercal f_s$
- An agent gets the same reward as the teacher if $\mathbb{E}_{s \sim p_\pi}[f_s] = \mathbb{E}_{s \sim \mathcal{D}}[f_s]$
  - This is also necessary, under mild conditions
- So let's optimize for the feature expectation $\max_\pi \mathbb{E}_{s \sim p_\pi}[\theta^\intercal f_s]$
  - But with what reward parameters?
- Idea: expert teacher should do max better than our learner on the true reward

$$\max_\theta (\mathbb{E}_{s \sim \mathcal{D}}[\theta^\intercal f_s] - \max_\pi \mathbb{E}_{s \sim p_\pi}[\theta^\intercal f_s])$$

Modeling bounded teachers

- We'd like to have an expert teacher who optimizes the return

$$\max_{\pi_T} \mathbb{E}_{\xi \sim p_T}[\theta^\intercal f_\xi] = \max_{\pi_T} \mathbb{E}_{s \sim p_T}[\theta^\intercal f_s]$$

- But suppose π_T has bounded ability to diverge from random behavior

$$\max_{\pi_T} \mathbb{E}_{\xi \sim p_T}[\theta^\intercal f_\xi] + \mathbb{H}[p_T]$$

- The optimal trajectory distribution satisfies

$$p_\theta(\xi) = \frac{1}{Z_\theta} p_0(\xi) \exp(\theta^\intercal f_\xi)$$

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

  - Approximation: ignore dynamical constraints that can make this unachiavable

MaxEnt IRL

$$p_\theta(\xi) = \frac{1}{Z_\theta} p_0(\xi) \exp(\theta^\intercal f_\xi)$$

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

- We now optimize the empirical log likelihood of demonstrations

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

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

- To compute the gradient, we need to take the forward expectation of p_θ

MaxEnt IRL — backward recursion

- Compute the partition function recursively backward

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

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

- Use it to compute local policy

$$\pi_\theta(a_t|s_t) = \pi_0(a_t|s_t)\frac{Z_{s_t,a_t;\theta}}{Z_{s_t;\theta}}$$

- Globally, this policy may be inconsistent $p_\theta \neq p_{\pi_\theta}$
  - but MaxEnt IRL uses it as an approximation

MaxEnt IRL

- Compute $Z_\theta = \mathbb{E}_{\xi \sim p_U}[\exp(\theta^\intercal f_\xi)]$ recursively backward
- Compute $\mathbb{E}_{\bar{\xi} \sim p_{\pi_\theta}}[f_{\bar{\xi}}]$ recursively forward
- Take a gradient step $\nabla_\theta \log p_\theta(\xi) = f_\xi - \mathbb{E}_{\bar{\xi} \sim p_{\pi_\theta}}[f_{\bar{\xi}}]$
- Repeat

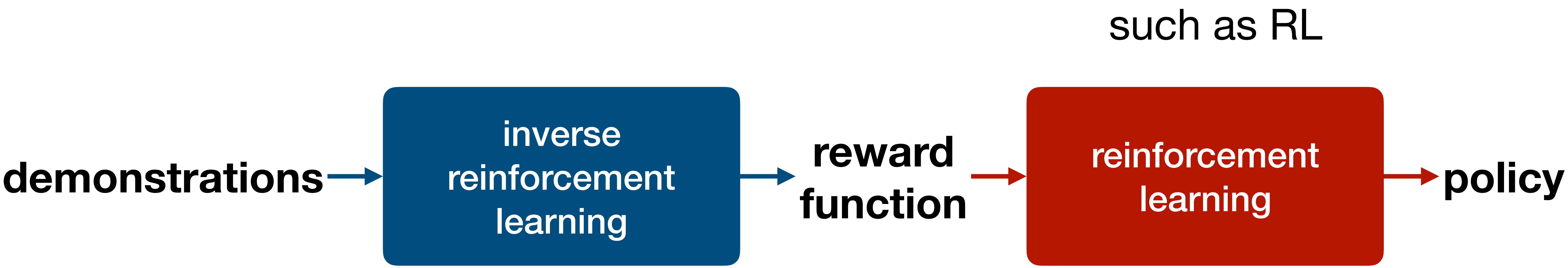
- At the optimum we have feature matching $\mathbb{E}_{\xi \sim \mathcal{D}}[f_\xi] = \mathbb{E}_{\xi \sim p_{\pi_\theta}}[f_\xi]$
- In fact, we have approximated $\max_\theta \mathbb{H}[\pi_\theta] \text{ s.t. } \mathbb{E}_{\xi \sim \mathcal{D}}[f_\xi] = \mathbb{E}_{\xi \sim p_{\pi_\theta}}[f_\xi]$

MaxEnt IRL limitations

- Approximation ignores dynamical constraints
- Policy estimation and visitation frequencies in each gradient step
- Model-based

IRL downstream tasks

- Our motivation: to learn a reward function for downstream tasks



- IL = RL ○ IRL
- But our algorithms go through learning a policy anyway
  - Let's optimize IRL for the overall IL tasks

IL as RL ○ IRL

- Entropy-regularized RL:

$$\max_{\pi \in \Pi} \mathbb{E}_{s \sim \bar{p}_\pi}[r(s)] + \mathbb{H}(\pi)$$

- MaxEnt IRL, with reward-function regularizer $\psi : \mathbb{R}^{\mathcal{S}} \to \mathbb{R}$:

$$\max_{r \in \mathbb{R}^{\mathcal{S}}} \mathbb{E}_{s \sim \bar{p}_T}[r(s)] - \max_{\pi \in \Pi}(\mathbb{E}_{s \sim \bar{p}_\pi}[r(s)] + \mathbb{H}(\pi)) - \psi(r)$$

- With respect to r, our objective is

$$\psi^*(\bar{p}_T - \bar{p}_\pi) = \max_{r \in \mathbb{R}^{\mathcal{S}}} (\bar{p}_T - \bar{p}_\pi) \cdot r - \psi(r)$$

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

Reward-function regularizers

$$\psi^*(\bar{p}_T - \bar{p}_\pi) = \max_{r \in \mathbb{R}^\mathcal{S}} (\bar{p}_T - \bar{p}_\pi) \cdot r - \psi(r)$$

- No regularizer $\psi = 0$ → solution only exists when $\bar{p}_T = \bar{p}_\pi$
  - This is really what we want, but challenging to solve
- Hard linearity constraint: $\psi(r) = \begin{cases} 0 & r(s) = \theta^\intercal f_s \\ \infty & \text{otherwise} \end{cases}$
  - Implies max-entropy feature matching (i.e. MaxEnt IRL)
  - Great when the reward function really is linear in f, otherwise no guarantee

Teacher-based reward-function regularizer

- Consider the regularizer

$$\psi_{\mathrm{GA}}(r) = \mathbb{E}_{s\sim\bar{p}_T}[r(s) - \log(1 - \exp(-r(s)))]$$

- It's convex conjugate is

$$\psi^*_{\mathrm{GA}}(\bar{p}_T - \bar{p}_\pi) = \max_{r\in\mathbb{R}^{\mathcal{S}}}(\bar{p}_T - \bar{p}_\pi)\cdot r - \psi(r)$$

$$= \max_{r\in\mathbb{R}^{\mathcal{S}}} \mathbb{E}_{s\sim\bar{p}_T}[r(s) - r(s) + \log(1 - \exp(-r(s)))] - \mathbb{E}_{s\sim\bar{p}_\pi}[r(s)]$$

- If we set $D(s) = \exp(-r(s))$

  then $\psi^*_{\mathrm{GA}}(\bar{p}_T - \bar{p}_\pi) = \mathbb{E}_{s\sim\bar{p}_\pi}[\log D(s)] + \mathbb{E}_{s\sim\bar{p}_T}[\log(1 - D(s))]$

Generative Adversarial Networks

- Focus the training of a generative model $p_\theta(s)$ on failure modes
- Also train a discriminator $D_\phi(s) \in [0,1]$ to score instances
  - If generated instances are like actions, then D_ϕ is like a critic
- $D_\phi(s)$ predicts the probability $p(\text{learner}|s) = \dfrac{p_\theta(s)}{p_\theta(s) + p_T(s)}$
- The discriminator can be trained with the cross-entropy loss

$$\max_\phi \mathbb{E}_{s\sim p_\theta}[\log D(s)] + \mathbb{E}_{s\sim p_T}[\log(1 - D(s))]$$

- The generator tries to fool the discriminator $\max_\theta \mathbb{E}_{s\sim p_\theta}[\log D(s)]$

Generative Adversarial Imitation Learning (GAIL)

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

repeat

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

take discriminator gradient ascent step

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

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

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

Recap

- To understand behavior, infer the intentions of observed agents
- If the teacher is optimized for a reward function
  - The reward function should be such that an optimizer behaves like the teacher
  - State (or state–action occupancy) of learner should match the teacher
- In this view, IRL is a game:
  - Reward is optimized to show how much better the teacher is than the learner
  - Policy is optimized to be good too
  - Reward is like a discriminator, policy like a generator

Control as inference

- Consider soft "success" indicators

$$p(v_t = 1 | s_t, a_t) = \exp \beta r(s_t, a_t)$$

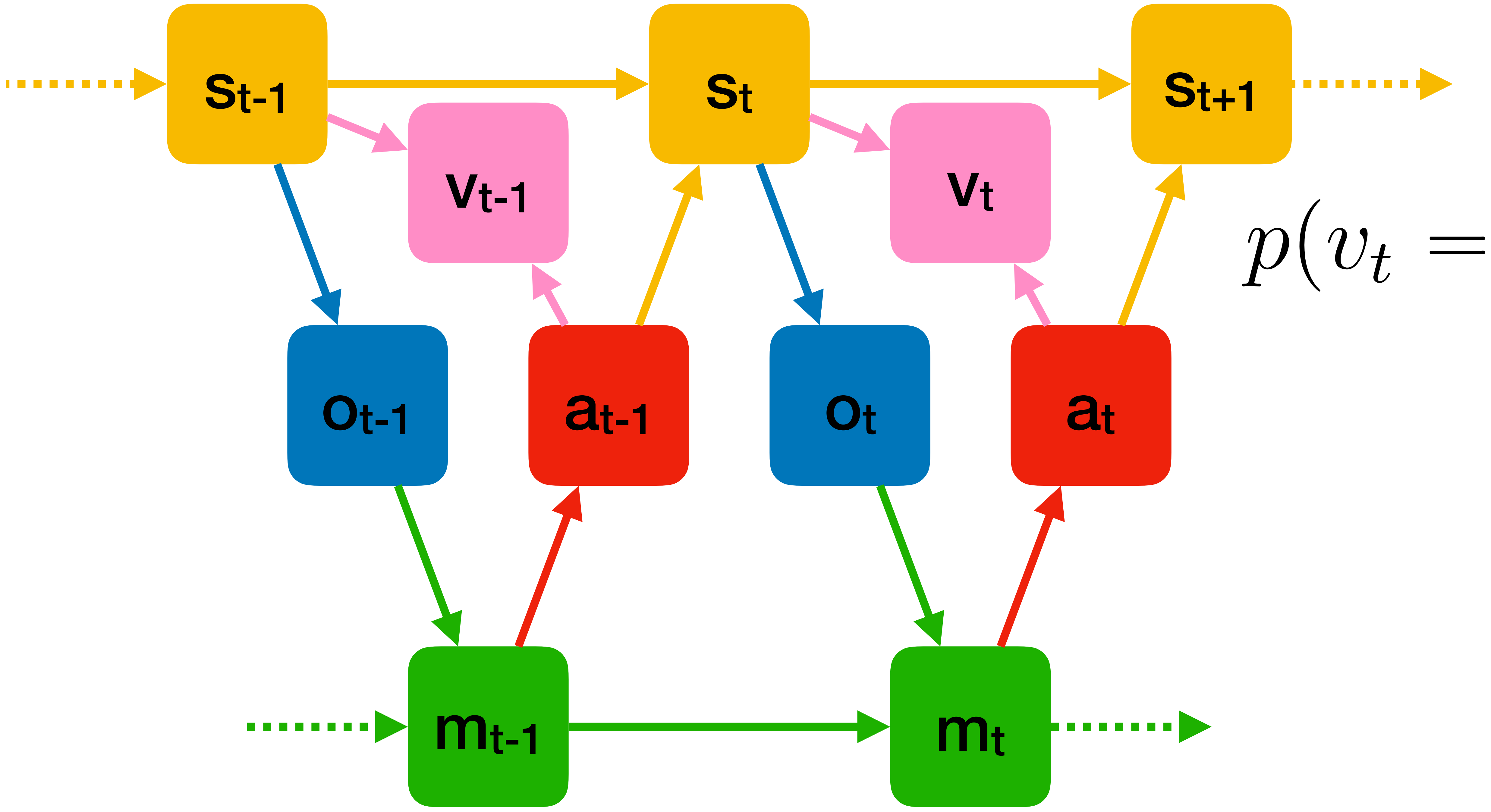
- What is the log-probability that an entire trajectory ξ "succeeds"?

$$\log p(\mathcal{V}|\xi) = \sum_t \log p(v_t = 1 | s_t, a_t) = \beta \sum_t r(s_t, a_t) = \beta R$$

- What is the posterior distribution over trajectories, <u>given success</u>?

$$p(\xi|\mathcal{V}) = \frac{p_0(\xi) p(\mathcal{V}|\xi)}{p_0(\mathcal{V})} = \frac{p_0(\xi) \exp \beta R}{Z}$$

Pseudo-observations



$$p(v_t = 1 | s_t, a_t) = \exp \beta r_t$$