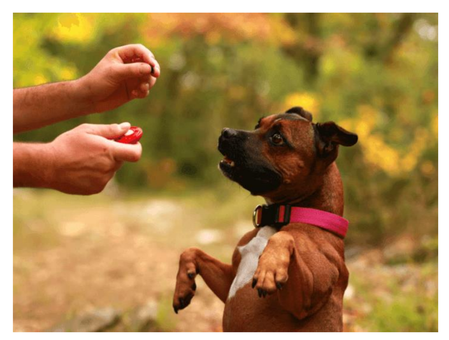
The Sample Complexity of Teaching-by-Reinforcement

Xuezhou Zhang[1], Shubham Kumar Bharti[1], Yuzhe Ma[1], Adish Singla[2] and Jerry Zhu[1]
[1]University of Wisconsin-Madison, [2]MPI-SWS

Problem Statement

Problem Setup:

- A teacher with full knowledge of the MDP wants to teach the optimal policy to the learner asap.
- Teacher can manipulate the transition/rewards.
- How many samples is required to learn the optimal policy with a teacher?



Prior Work:

1. Teaching by demonstration (imitation learning):
   - well-studied.
   - Sample complexity: $O\left(\frac{H^2 S}{\epsilon}\right)$ to learn an ϵ-optimal policy.
2. Teaching by reinforcement (Reward Shaping):
   - Performs well in practice.
   - Little theoretical understanding.

Questions we answer in this work:

- What is the best teaching strategy given different teacher control?
- What is the sample complexity of learning under optimal teaching?

Sample Complexity of Teaching

Level 1	Level 2	Level 3	Level 4
none	keep a_t	$s_{t+1} : P(s_{t+1} \mid s_t, a_t) > 0$	$s_{t+1} \sim P(\cdot \mid s_t, a_t)$
S	$S(A-1)$	$O\left(SAH\left(\frac{1}{1-\epsilon}\right)^D\right)$	$O\left(SAH\left(\frac{1}{(1-\epsilon)p_{\min}}\right)^D\right)$

Table 1: Summary of Main Results

Definition 1 *We define the* ***minimum teaching length*** *as*

$$\mathrm{METaL}(M, Q_0, \pi^\dagger) = \min_{T,(s_t,a_t,r_t,s_{t+1})_{0:T-1}} \mathbb{E}[T], \ s.t.\ \pi_T = \pi^\dagger,$$

where the expectation is taken over the randomness in the MDP M (transition dynamics) and the learner L (stochastic behavior policy).

Definition 2 *The* ***teaching dimension*** *of an RL learner L w.r.t. a family of MDPs $\mathcal{M}$ is defined as the worst-case METal:*

$$TDim = \max_{\pi^\dagger, Q_0, M \in \mathcal{M}} \mathrm{METaL}(M, Q_0, \pi^\dagger).$$

Definition. *Let the* ***diameter*** *D of an MDP be defined as the minimum path length to reach the hardest-to-get-to state in the underlying directed transition graph of the MDP. Specifically,*

$$D = \max_{s \in S} \min_{T,(s_0,a_0,s_1,a_1,\dots,s_T=s)} T, \ s.t.\ \mu_0(s_0) > 0, P(s_{t+1}|s_t,a_t) > 0, \forall t$$

Definition. *Let the* ***minimum transition probability*** *$p_{\min}$ of an MDP be defined as*

$$p_{\min} = \min_{s,s' \in \mathcal{S}, a \in \mathcal{A}, P(s'|s,a)>0} P(s'|s,a).$$

4 Levels of Teaching

Level 1 Teacher

- can generate arbitrary (s_t, r_t, s_{t+1}), and override the agent action a_t.
- None of these need to obey the MDP (specifically μ_0, P, R).
- Sample Complexity: S
- How: Give $(s, \pi^*(s)$, big reward) for each $s \in S$.

Level 2 Teacher

"Cost of free will" = A

- can generate arbitrary (s_t, r_t, s_{t+1}), but can't override action a_t.
- Sample Complexity: $S(A-1)$
- How: Now each state needs to be visited at least A-1 times to learn the desired action.

Level 3 Teacher

"Cost of navigation" = H

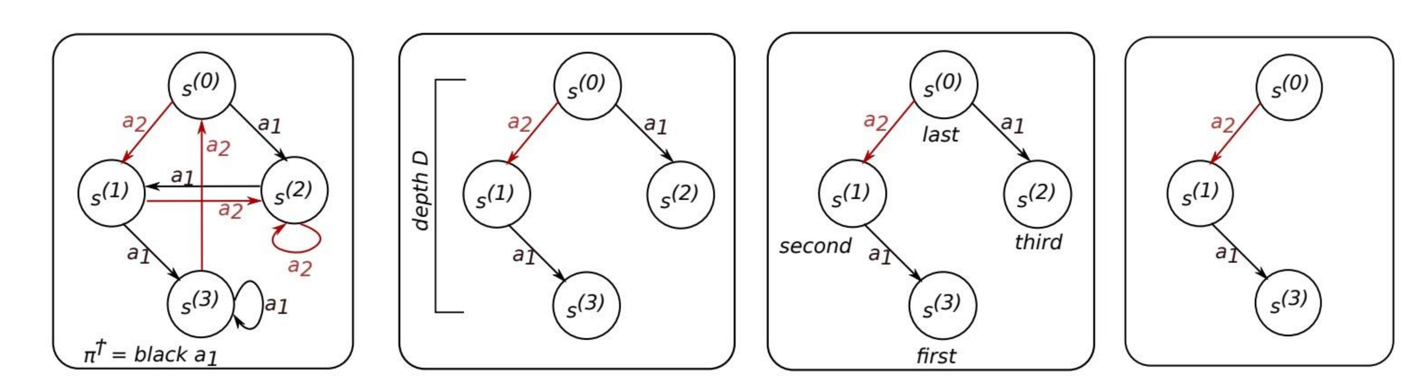
- can generate arbitrary r_t.
- but can only generate MDP-supported initial state and next state, i.e. $\mu_0(s_0) > 0$, and $P(s_{t+1}|s_t, a_t) > 0$.
- Sample Complexity: $\Theta\left(\mathrm{SAH}\left(\frac{1}{1-\epsilon}\right)^D\right)$.

Level 4 Teacher

"Cost of no simulator" = p_{min}^{-D}

- can generate arbitrary r_t.
- must obey the MDP transition, i.e. $s_0 \sim \mu_0(s_0)$, and $s_{t+1} \sim P(\cdot|s_t, a_t)$.
- Sample Complexity:$\Theta\left(\mathrm{SAH}\left(\frac{1}{p_{\min}(1-\epsilon)}\right)^D\right)$.

- **NavTeach Algorithm for Level 3 & 4 Teaching**



(a) MDP (b) Breadth-First Tree (c) Depth-First Traversal (d) Navigation Policy

1. Define an order of the states to teach.
2. For the current state, teach a navigation path to that state, and then teach the target action in that state.

Key challenge: teach in an such an order that the target action in earlier states wouldn't interfere with the navigation to later states.

Our solution: Construct a breath-first tree T on the underlying graph. Teach in the order of depth-first traversal of T.

Preliminaries

Episodic MDP:

- The environment is an episodic MDP $\mathcal{M} = (S, A, R, P, \mu_0, H)$:
  - S is the state space.
  - A is the action space.
  - $R: S \times A \to \mathbb{R}$ is the reward function.
  - $P: S \times A \to \Delta_S$ is the transition function.
  - $\mu_0 \in \Delta_S$ is the initial state distribution.
  - H is the episode length.
- The learner wants to learn the optimal policy:

$$\pi^* = \arg\max_{\pi: S \to A} \mathbb{E}_M \left[\sum_{h=1}^{H} R(s_h, \pi(s_h))\right]$$

ϵ-Greedy Q-Learner:

- The agent performs standard Q-learning, defined by

$$Q_{t+1}(s_t, a_t) \leftarrow (1 - \alpha_t) Q_t(s_t, a_t) + \alpha_t \left(r_t + \gamma \max_{a' \in A} Q_t(s_{t+1}, a')\right)$$

where α_t is the learning rate and γ is the optional discounting factor.

- The agent behaves according to the ε-greedy policy

$$a_t \leftarrow \begin{cases} \arg\max_a Q_t(s_t, a), & \text{w.p. } 1-\epsilon \\ \text{uniform from } A, & \text{w.p. } \epsilon. \end{cases}$$

Contact

Xuezhou Zhang
University of Wisconsin, Madison
zhangxz1123@cs.wisc.edu