687 2017-12-12

Inverse RL

RL: Given samples S, A, P, R, d_0, γ find π^*

IRL: " " $S, A, P, d_0, \gamma, \pi^*$ find R

↳ E.g., learning from demonstration (LFD)

- Abstraction selection

e.g.

$R_t = w^T \phi(S_T)$

Multiagent RL

Stochastic game generalization of an MDP

$(\mathcal{S}, \mathcal{A}_1, \mathcal{A}_2, \ldots, \mathcal{A}_n, R_1, R_2, \ldots, R_n, d_0, \gamma)$

$P(S, a_1, a_2, \ldots, a_n, S') = \Pr(S_{t+1} = S' \mid S_t = s, A_1 = a_1, \ldots, A_n = a_n)$

$R_i(S, a_1, a_2, \ldots, a_n, S')$

- Fully cooperative if $\forall i, j,\ R_i = R_j$
- Otherwise, need to employ Game Theory → equilibria

Pareto Frontier + multi-objective optimization

Let $f_1, \ldots, f_n$ be n objectives, where each $f_i : \mathcal{X} \Rightarrow \mathbb{R}$

The Pareto frontier is a set of solutions:

$$P \triangleq \{x \in \mathcal{X} : \forall x' \in \mathcal{X} : (\underbrace{\exists i : f_i(x') > f_i(x)}_{\text{making } f_i \text{ better}}) \Rightarrow (\underbrace{\exists j : f_j(x') < f_j(x)}_{\text{makes } f_j \text{ worse}})\}$$

RL Theory

PAC-MDP : Probably approximately correct in MDPs

(To probability $1-\delta$, after a fixed number of timesteps less than a polynomial in $|\mathcal{S}|$, $|\mathcal{A}|$, $1/\epsilon$, $1/\delta$, and $\frac{1}{1-\gamma}$, it returns a policy whose expected return is within ϵ of $J(\pi^*)$.)

- Sample complexity : the polynomial fn. [Kakade 2003]

Hoeffding's Inequality

Let $x_1, x_2, \dots, x_n$ be n i.i.d. random variables s.t $\mu = [x_i]$ and $x_i \in [a,b]$ (or $Pr(x_i \in [a,b]) = 1$)

then

$$Pr\left(\mu \geq \frac{1}{n}\sum_{i=1}^{n} x_i - (b-a)\sqrt{\frac{\ln 1/\delta}{2n}}\right) \geq 1-\delta$$

↓

Theory: 4x4, 5x5 grid world Lattmore/Hutter 2015

↳ 10^{11} steps to guarantee within 10% to high prob.

Kearning (sp?) + Singh 1998

Regret

The regret of an algorithm over $T = K \cdot L$ time steps
(some constant; horizon)

is $\text{Regret}(K) \triangleq \sum_{k=1}^{K} V^*(S_k) - V^{\pi_k}(S_k)$

UCB + Thompson Sampling

- Bandit problem: Always choose bandits w/ higher upper bounds (e.g., via Hoeffding's)
- Sample according probability distributions that you update as you go. Ian Osband

Lower bound: $\Omega\left(\sqrt{HSAT}\right)$
↳ L = horizon

Azar, Osband, Munos: $\tilde{O}\left(\sqrt{HSAT} + H^2S^2A + H\sqrt{T}\right)$
(upper bound, ie., an algorithm)

Off-policy (policy) evaluation

$D = \{H_i\}_{i=1}^{n}$ $H : \pi_b \leftarrow$ behavior policy

Goal: estimate $J(\pi_e)$ given D, π_b

Method 1: model-based

Method 2: sampling-based: Importance-sampling

$$\hat{J}(\pi_e, D) = \frac{1}{n} \sum_{i=1}^{n} \frac{Pr(H_i \mid \pi_e)}{Pr(H_i \mid \pi_b)} g(H_i)$$

$g(H_i) \leftrightarrow$ discounted sum of rewards

adjusts for likelihood of different histories H_i under different policies.

$$= \frac{1}{n} \sum_{i=1}^{n} \frac{d_0(S_0)\pi_e(a_1|S_1)P(s_1,a_1,s_2)P(R_1)\cdots}{d_0(S_0)\pi_b(a_1|S_1)P(s_1,a_1,s_2)P(R_1)\cdots} g(H_i)$$

$$= \frac{1}{n} \sum_{i=1}^{n} \frac{\pi_e(a_1|S_1)\pi_e(a_2|S_2)\cdots}{\pi_b(a_1|S_1)\pi_b(a_2|S_2)\cdots} g(H_i)$$

Batch RL - works from a batch of data from one policy
→ get a better policy

Safe RL - all policy moves are improvements

Average reward RL - average, not discounted, reward
Sridhar Mahadevan

Deep Learning RL

DL → gives a function approximation space

RL → how to train a fn. approx. for sequential decision problems (MDPs)

Q learning + Convolutional neural net + experience replay
→ tweaked to use target net

Dueling networks 199x Baird's Advantage Updating

→ Atari