Policy gradient methods for reinforcement learning with function approximation

NIPS

Richard S. Sutton, David McAllester, Satinder Singh, Yishay Mansour AT&T Labs

TinRay

Tasks Suitable for Reinforcement Learning

- Many action, then get reward.
- Such as game playing, go[1], etc.

[1]Should go be capitalized?

Markov Decision Process

- Environment:
  - Set of all states: $\mathcal{S}, \mathcal{S}^+$
  - Set of all actions possible in state $\mathcal{A}(s)$.
  - Set of all possible rewards: $\mathcal{R}$
  - Transition probability: $p(s'|s, a)$
- Agent:
  - Policy: $\pi(a, s) \in \mathbb{R}$, $\pi(s) \in \mathcal{A}$
- Trajectory: $s_0, a_0, s_1, a_1, \cdots$
- Episode: $s_0, a_0, r_0, s_1, a_1, r_1, \cdots$

From supervised learning to RL

- Supervised Learning
  - Train Data: $x_1, x_2, \cdots, y_1, y_2, \cdots$.
  - To learn: f that map x to y.
- Reinforcement Learning
  - Environment
  - Find a way to maximize total reward.

Two perspectives toward RL

- Actor: Learn what to do given a state s.
  - But we don' t know optimal action for each state s.
  - So policy gradient is there.
- Critic: Learn how good an action a is given a state s.
  - But we don' t have truth value of (s, a).
  - So we have Monte Carlo, TD methods here.

Outline

- Critic
  - Monte Carlo
    - Monte Carlo Estimation of Action Values
    - Monte Carlo Control
    - ϵ-Greedy
    - Off-policy Prediction via Importance Sampling
  - Time Difference
    - TD Prediction
    - SARSA
    - Q-Learning
- Actor
  - REINFORCE
  - Policy-Gradient (Actor-Critic?)

Goal of learning critic

- How good an action a given s is?
  Ans: Expected total reward after action a done on state s with policy π.

$$q_\pi(s,a) = E\left\{\sum_{k=1}^{\infty} \gamma^{k-1} r_{t+k} | s_t = s, a_t = a, \pi\right\}$$

- So we want to learn $Q_\pi(s,a)$ to estimate $q_\pi(s,a)$, that is, minimize

$$(Q_\pi(s,a) - \hat{q}_\pi(s,a))^2$$

  where $\hat{q}_\pi$ is an (unbiased) estimator of q_π.
- $\hat{q}_\pi$ is actually the difference between MC and TD.

Monte Carlo Estimation of Action Values

- Given a policy $\pi(s, a)$, how to estimate $q_\pi(s, a)$?
- We can generate many episodes with π

$$s_0^0, a_0^0, r_0^0, s_1^0, a_1^0, r_1^0, \cdots$$
$$s_0^1, a_0^1, r_0^1, s_1^1, a_1^1, r_1^1, \cdots$$
$$\vdots$$

Then for each (s, a) pair $\hat{q}_\pi$ is average total reward get after action a is taken on state s in those episodes.

$$\hat{q}_\pi(s, a) = \frac{1}{|\{(e, t)|s_t^e = s\}|} \sum_{\{(e,t)|s_t^e = s\}} \text{Return}(s_t^e, a_t^e)$$

where $\text{Return}(s_t^e, a_t^e)$ is defined as

$$\text{Return}(s_t^e, a_t^e) = \sum_{k=0} \gamma^k r_{t+k}^e$$

Monte Carlo Estimation of Action Values - Example

Monte Carlo Control

- How about random initialize a policy π_0;
- then estimate $\hat{q}_{\pi_0}(s, a)$ with Monte Carlo method;
- then make a new policy π_1 base on $\hat{q}_{\pi_0}$

$$\pi_1(s) = \arg\max_a \hat{q}_{\pi_0}(s, a)$$

  which is better than π_0
- then estimate $\hat{q}_{\pi_1}(s, a)$ with Monte Carlo method;
- then make a new better policy...
- And so on...
- So we can get opitmal π!!

ϵ Greedy Method

- Wait! There is nearly no exploration!
- So we can add probability ϵ to take action randomly.

$$\begin{cases} \pi_k(s,a) = \frac{\epsilon}{|(A)|} + 1 - \epsilon & a = \arg\max_a \hat{q}_{\pi_{k-1}}(s,a) \\ \pi_k(s,a) = \frac{\epsilon}{|(A)|} & \text{otherwise} \end{cases}$$

Off-policy Prediction via Importance Sampling

- How about using another policy μ to do exploration?
- Original Mote Carlo use average return in many episodes to estimate

$$
\begin{aligned}
q_\pi(s,a) =& E_{\sim\pi}\{\text{Return}(s,a)\} \\
=& E\left\{\sum_{k=1}\gamma^{k-1} r_{t+k} | s_t = s, a_t = a, \pi\right\} \\
=& (\sum_{k=1}\gamma^{k-1} r_{t+k}) Pr(r_t, s_{t+1}, a_{t+2}, r_{t+2}, \cdots) \\
=& (\sum_{k=1}\gamma^{k-1} r_{t+k}) p(s_k, r_{k-1}|a_{k-1}) \prod_{k=1} \pi(s_k, a_k) p(s_{k+1}, r_k|a_k) \\
=& (\sum_{k=1}\gamma^{k-1} r_{t+k}) p(s_k, r_{k-1}|a_{k-1}) \prod_{k=1} \frac{\pi(s_k, a_k)}{\mu(s_k, a_k)} \prod_{k=1} \mu(s_k, a_k) p(s_{k+1}, r_k|a_k) \\
=& E_{\sim\mu}\left\{\text{Return}(s,a) \prod_{k=1} \mu(s_k, a_k)\right\}
\end{aligned}
$$

Off-policy Prediction via Importance Sampling - Cont.

- Then we can estimate q_π with policy μ by

$$\hat{q}_\pi(s,a) = \frac{1}{|\{(e,t)|s_t^e = s\}|} \sum_{\{(e,t)|s_t^e=s\}} \text{Return}(s_t^e, a_t^e) \prod_{k=1} \mu(s_k, a_k)$$

- Estimating expectation from different distribution is what "importance sampling" does.