Summary of Policy Gradient Methods

Monday, December 28, 2020 10:57 PM

REINFORCE increases probability of "good actions" and decreases probability of "bad actions"

What are Policy Gradient Methods?

- **Policy-Based Methods** are a class of algorithms that search directly for the optimal policy, w/o simultaneously maintaining value function estimations (Hill Climbing, Steepest Ascent, Hill Climbing, Cross Entropy Method)

- **Policy-Gradient Methods** are a subclass of policy-based methods that estimate the weights of an optimal policy through gradient ascent

- In this lesson, we represent the policy

- In this ... as a NN where the goal to find the optimal weights θ that maximizes the rewards.

The Big Picture

- Policy Gradient Method will iteratively amend the policy network weights to:
  - → Make (state, action) pairs that resulted in positive return more likely
  - → Make (state, action) pairs that resulted in negative return less likely.

Problem Setup

- The trajectory τ is a state-action sequence $s_0, a_0, \cdots, s_H, a_H, s_{H+1}$

- In this lesson, we will use notation $R(\tau)$ to refer to the return to corresponding trajectory.

- Goal: Find θ of the policy network that will maximize expected return.

$$U(\theta) := \sum_{\tau} \mathbb{P}(\tau; \theta) R(\tau)$$

REINFORCE

- PSEUDOCODE:

1. Use the policy π_θ to collect m trajectories $\{\tau^{(1)}, \tau^{(2)}, \ldots \tau^{(m)}\}$ with horizon H. We refer to i-th trajectory as

$$\tau^{(i)} = (s_0^{(i)}, a_0^{(i)}, \ldots, s_{H+1}^{(i)}, a_H^{(i)}, s_{H+1}^{(i)})$$

2. Use trajectories to estimate gradient:

$$\nabla_\theta U(\theta) \approx \hat{g} := \frac{1}{m} \sum_{i=1}^{n} \sum_{t=0}^{H} \nabla_\theta \log \pi_\theta(a_t^{(i)} | s_t^{(i)}) R(\tau^{(i)})$$

3. Update weights of policy

$$\theta \leftarrow \theta + \alpha \hat{g}$$

4. Loop over 1-3

Derivation

- we derived the **likelihood ratio policy gradient:**

$$\nabla_\theta U(\theta) = \sum_\tau P(\tau;\theta) \nabla_\theta \log P(\tau;\theta) R(\tau)$$

- We can approximate the gradient above w/ a

sample-weighted average:

$$\nabla_\theta U(\theta) \approx \frac{1}{m} \sum_{i=1}^{m} \nabla_\theta \log P(\tau^{(i)}; \theta) R(\tau^{(i)})$$

- we calculated the following:

$$\nabla_\theta \log P(\tau^{(i)}; \theta) = \sum_{t=0}^{H} \nabla_\theta \log \pi_\theta(a_t^{(i)} | s_t^{(i)})$$

What's Next?

- REINFORCE can solve Markov Decision Processes (MDP) w/ either discrete or Continuous Action Spaces.