Homework 5

Due on December 18th

The homework involves concepts surrounding the policy gradient theorem. Consider a discounted Markov decision process (X, A, P, γ, r) with finite state and action spaces $\mathcal{X}$ and $\mathcal{A}$, transition function P, discount factor $\gamma \in (0, 1)$, and reward function r. Let π_θ be a parametrized stochastic policy with $\pi_\theta(a|x)$ being the probability of taking action a in state x. In particular, consider the policy

$$\pi_\theta(a|x) = \frac{e^{\theta^\top \phi(x,a)}}{\sum_{b \in \mathcal{A}} e^{\theta^\top \phi(x,b)}}, \tag{1}$$

where $\phi : \mathcal{X} \times \mathcal{A} \to \mathbb{R}^d$ is a d-dimensional feature vector and $\theta \in \mathbb{R}^d$ is a d-dimensional vector of parameters. Assuming that ϕ is bounded, we can apply the following corollary of the policy gradient theorem:

Corollary 1. *Let $h : \mathcal{X} \to \mathbb{R}$ be an arbitrary function. Assuming that $\pi_\theta(a|x) > 0$ for all (x, a) and that $\nabla_\theta \log \pi_\theta(a|x)$ exists, the gradient of $\rho(\theta) = V^{\pi_\theta}(x_0)$ can be written as*

$$\begin{aligned}\nabla_\theta \rho(\theta) &= \sum_{(x,a) \in \mathcal{X} \times \mathcal{A}} \mu_\theta(x) \pi_\theta(a|x) \nabla_\theta \log \pi_\theta(a|x) \left(Q^{\pi_\theta}(x, a) - h(x)\right) \\ &= \mathbb{E}_{X \sim \mu_\theta, A \sim \pi_\theta(\cdot|X)} \left[\nabla_\theta \log \pi_\theta(A|X) \left(Q^{\pi_\theta}(X, A) - h(X)\right)\right].\end{aligned}$$

A useful consequence of the above corollary is that one come construct an unbiased estimate g of the policy gradient by plugging in an unbiased estimate of $Q^{\pi_\theta}(X, A)$, and replacing the expectation above by a single sample (or an average of a number of samples). For instance, REINFORCE estimates Q^{π_θ} by Monte Carlo rollouts and uses several samples to approximate the expectation. The goal of the homework is to implement a gradient-ascent algorithm based on these ideas. The first question is about a key component for estimating the gradient.

Question 1: Compute the gradient $\nabla_\theta \log \pi_\theta(a|x)$ for the policy given in Equation (1)!

The second question is concerned with a concrete MDP and computing estimates of the policy gradient via the above corollary.

Question 2: Consider an MDP with a single state x (that is, $\mathcal{X} = \{x\}$) and two actions: a_1 and a_2 (so that $\mathcal{A} = \{a_1, a_2\}$). In the MDP, the two actions lead to 0/1 valued random rewards such that each action a_i yields a reward of 1 with probability p_i, with $p_1 = 1/2$ and $p_2 = 1/2 + \Delta$ for some $\Delta > 0$. The task is to implement a policy-gradient-based learning algorithm that repeats the following steps in each round $k = 0, 1, \ldots$:

1. Compute policy π_{θ_k} and draw action $A_k \sim \pi_{\theta(k)}(\cdot|x)$.

2. Let I_k be the index of action A_k (i.e., 1 if $A_k = a_1$ and 2 otherwise) and observe reward

$$R_k = \begin{cases} 1, & \text{with probability } p_{I_k}, \\ 0, & \text{otherwise.} \end{cases}$$

3. Based on Corollary 1 and the observed reward, compute an estimate g_k of the policy gradient.
4. Update the parameter vector as

$$\theta_{k+1} = \theta_k + \alpha_k g_k.$$

Specifically, the task is to answer the following questions:

Question 2.1: How does the choice of γ influence the behavior of the algorithm?

Question 2.2: Fix $\gamma = 0.99$ and $\Delta = 0.05$, and consider the step sizes

- $\alpha_k = c/\sqrt{k}$,
- $\alpha_k = c/k$, and
- $\alpha_k = c/k^2$

for various choices of c. Plot $\pi_{\theta_k}(a_2|x)$ as a function of k.

Question 2.3: Pick the best of these step sizes and plot $\pi_{\theta_k}(a_2|x)$ as a function of k for $\Delta \in \{0.01, 0.05, 0.1, 0.5\}$.

Hints: First observe that $\mu_\theta(x) = 1$. The key challenge is coming up with a useful definition of g_k. To this end, it is suggested to use the second line in the corollary and estimate the expectation by a single sample A_k and use

$$\widehat{Q}^{\pi_{\theta_k}}(x, A_k) = R_k + \gamma V^{\pi_{\theta_k}}(x)$$

to estimate $Q^{\pi_{\theta_k}}(x, A_k)$. Using Corollary 1 with $h(x) = \gamma V^{\pi_{\theta_k}}(x)$ yields a particularly simple way of computing g_k.

As for the choice of ϕ, it is recommended to use the features

$$\phi_i(x, a) = \mathbb{1}_{\{a=a_i\}} = \begin{cases} 1, & \text{if } a = a_i, \\ 0, & \text{otherwise.} \end{cases}$$

for $i = 1, 2$.