Robot Learning **Assignment 2**
CS 4756 Spring 2025 **Due 11:59pm Friday, February 28**

(1) Policy Gradients (10 points)

Recap: Recall that the goal of RL is to learn some θ^* that maximizes the objective function:

$$J(\theta) = \mathbb{E}_{\tau \sim \pi_\theta(\tau)} \big[r(\tau) \big] \tag{1}$$

where each τ is a rollout of length T_τ and $r(\tau) = \sum_{t=0}^{T_\tau - 1} r(s_t, a_t)$ is the reward for that rollout. $\pi_\theta(\tau)$ is the probability of the rollout under policy π_θ, i.e. $\pi_\theta(\tau) = \Pr[s_0]\pi_\theta(a_0|s_0) \prod_{t=1}^{T_\tau - 1} \Pr[s_t|s_{t-1}, a_{t-1}]\pi_\theta(a_t|s_t)$.

The policy gradient approach requires that we take the gradient of this objective as follows:

$$\begin{aligned} \nabla_\theta J(\theta) &= \nabla_\theta \int \pi_\theta(\tau) r(\tau)\ d\tau = \int \pi_\theta(\tau) \nabla_\theta \log \pi_\theta(\tau) r(\tau)\ d\tau && (2) \\ &= \mathbb{E}_{\tau \sim \pi_\theta(\tau)} \big[\nabla_\theta \log \pi_\theta(\tau) r(\tau) \big] && (3) \end{aligned}$$

The gradient can further be refined by noting that future actions do not affect past rewards (the causality assumption), resulting in the following "reward-to-go" formulation:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim \pi_\theta(\tau)} \left[\sum_{t=0}^{T_\tau - 1} \left(\nabla_\theta \log \pi_\theta(a_t|s_t) \cdot \sum_{t'=t}^{T_\tau - 1} r(s_{t'}, a_{t'}) \right) \right] \tag{4}$$

In this question, we consider a toy MDP and get familiar with computing policy gradients.

(a) Show the following step in 2 holds true and explain why this step is valid.:

$$\nabla_\theta \int \pi_\theta(\tau) r(\tau)\ d\tau = \int \pi_\theta(\tau) \nabla_\theta \log \pi_\theta(\tau) r(\tau)\ d\tau$$

(b) Starting from Equation 3, use the causality assumption (that future actions do not affect past rewards) to derive the "reward-to-go" formulation given in Equation 4. Show the intermediate steps.

(c) We introduce a baseline to reduce the variance of the policy gradient estimator $b(s_t)$, leading to the advantage function:

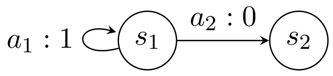
$$A^{\pi_\theta}(s_t, a_t) = Q^{\pi_\theta}(s_t, a_t) - V^{\pi_\theta}(s_t)$$

Prove that subtracting a baseline does not introduce bias in the gradient estimation, i.e., show that:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim \pi_\theta(\tau)} \left[\sum_{t=0}^{T_\tau - 1} \nabla_\theta \log \pi_\theta(a_t|s_t) \cdot A(s_t, a_t) \right] \tag{5}$$

Explain why the expectation of the baseline term vanishes.
Hint:

- Show that adding $b(s_t)$ does not affect the expected gradient by proving its expectation is zero.
- Rewrite the policy gradient with the bias and show that it does not change the expectation



Now, consider the following infinite-horizon MDP.
The initial state is always s_1, and the episode terminates when s_2 is reached. The agent receives reward 1 for taking action a_1 and reward 0 for taking action a_2. In this case, we can define the policy with a single parameter θ:

$$\pi_\theta(a_1|s_1) = \theta, \quad \pi_\theta(a_2|s_1) = 1 - \theta$$

$J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}[R(\tau)]$, and by the policy gradient method,

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau\sim\pi_\theta}\left[(\nabla_\theta \log \pi_\theta(\tau)) \cdot R(\tau)\right]$$

Here, τ is a trajectory $s_0, a_0, ..., s_T, a_T$ that terminates after $T + 1$ steps.
And, $R(\tau) = \sum_{t=0}^{T} r(s_t, a_t)$, and

$$\pi_\theta(\tau) = p(s_0)\pi_\theta(a_0|s_0)\prod_{t=1}^{T} p(s_t|s_{t-1}, a_{t-1})\pi_\theta(a_t|s_t).$$

(d) Using the policy gradient theorem and the definition of our policy, compute the gradient of the expected return of π_θ with respect to the parameter θ (Eq. 3). Do not use discounting. The sum should telescope to a closed form solution.
You may find this fact useful:

$$\sum_{k=1}^{\infty} k\alpha^{k-1} = \frac{d}{d\alpha}\sum_{k=1}^{\infty} \alpha^k$$

(e) Compute the expected return of the policy π_θ directly using the law of total expectation(Eq. 1). Compute the gradient of this expression and verify that it matches your result in **(d)**.

(f) Reward-to-go can be helpful and improve the statistical qualities of our policy gradient. Apply reward-to-go as an advantage estimator. Write the new policy gradient (Eq. 4), and verify that it is unbiased.

(2) Reward Shaping with an Approximate Value Function (5 points)

Previously, we saw how acting greedily with an approximate value function can result in a *worse* policy. Instead, what if we use the approximate value function to *shape the reward*?

Consider an infinite-horizon Markov Decision Process (MDP) with discount factor γ. We have a reference policy π with a true value function $V^{\pi}(s)$. To approximate this function, we collect rollouts using π and fit a neural network to estimate the value function, yielding $\hat{V}(s) \approx V^{\pi}(s)$, with an approximation error bounded by ϵ. Formally, we assume:

$$||V^{\pi} - \hat{V}||_{\infty} \leq \epsilon.$$

Now, we define a *reward bonus* using this approximate value function:

$$F(s, s') = \gamma \hat{V}(s') - \hat{V}(s)$$

This additional reward is given whenever the agent transitions from state s to state s'. Informally, it provides a small intermediate reward for moving toward states of higher estimated value. By adding these intermediate rewards, we aim to accelerate policy convergence, especially in environments with sparse rewards.
At each step i of an episode, the shaped reward R_i is then defined as

$$R_i = r_i + F(s_i, s_{i+1})$$

where r_i is the base reward $r(s_i, a_i)$ received for step i. We continue to use a *discount factor* of γ in computing total reward. Recall that for an infinite-horizon setting, the total cumulative reward is typically expressed as:

$$R = \sum_{i=0}^{\infty} \gamma^i r_i$$

In this problem, we will explore how this changes when shaping the reward with the approximate value function.

(a) Consider a given episode of potentially infinite-length, of visited states $s_0, s_1, ...$
Write out the total reward received in the shaped environment, expressed in terms of the total reward that would have been accrued in the unshaped environment. What is noticeable about this relationship?

(b) The policy $\hat{\pi}$ is found by optimizing the shaped rewards, while the policy π^* is found by optimizing the unshaped rewards. Although the policies are derived using different reward structures, we ultimately want to compare their performance using the same value function.

Explain why the performance of the optimal policy $\hat{\pi}$ computed with the shaped rewards is the same as the performance of the optimal policy π^* computed with the unshaped rewards. Specifically, explain why:

$$||V^{\pi^*} - V^{\hat{\pi}}||_{\infty} = 0$$

Hint: Use your interpretation from part (a) to reason about why the performance is unaffected by reward shaping. You can either use the math or provide an explanation based on the takeaway from part (a).

(3) (Mandatory for 5756): Off Policy Gradient Estimation (10 points)

In this problem, you will work towards deriving the off-policy gradient of a policy using importance weighting techniques. Consider a finite-horizon MDP $\mathcal{M} = \{\mathcal{S}, \mathcal{A}, P, R, H, \mu_0\}$, with a policy π_θ that we want to optimize, and a different policy π' that generates the trajectory data.

The trajectory $\tau = \{s_0, a_0, s_1, a_1, \ldots, s_{H-1}, a_H\}$ is sampled using the policy π', and the objective is to compute the gradient of π_θ using this off-policy data. The reward function is defined as the sum of rewards over the trajectory, $R(\tau) = \sum_{t=0}^{H-1} r(s_t, a_t)$, and the goal is to maximize the expected cumulative reward. Recall that the gradient of the objective function can be expressed as:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim \pi_\theta} \left[\nabla_\theta \log \pi_\theta(\tau) R(\tau)\right]$$

where $\pi_\theta(\tau) = \mu_0(s_0) \prod_{t=0}^{H-1} \pi_\theta(a_t|s_t) P(s_{t+1}|s_t, a_t)$.

Since the data is collected under a different policy π', we need to use importance weighting to correct for the distribution mismatch. This will allow us to estimate the policy gradient for π_θ using data collected from π'.

(a) Show that the policy gradient can be written using importance weights as:

$$\nabla_\theta J(\theta) = \mathbb{E}_{\tau \sim \pi'} \left[\frac{\pi_\theta(\tau)}{\pi'(\tau)} \nabla_\theta \log \pi_\theta(\tau) R(\tau)\right]$$

Hint: Multiply and divide by $\pi'(\tau)$, and use the fact that the expectation over π_θ can be transformed into an expectation over π' using importance sampling.

(b) Derive an expression for the ratio $\frac{\pi_\theta(\tau)}{\pi'(\tau)}$ in terms of the individual action probabilities $\pi_\theta(a_t|s_t)$ and $\pi'(a_t|s_t)$ for $t = 0, \ldots, H-1$.

Hint: Use the fact that the probability of a trajectory is the product of action probabilities and transition probabilities under the respective policies.

(c) Now, derive an expression for $\nabla_\theta \log \pi_\theta(\tau)$ in terms of $\nabla_\theta \log \pi_\theta(a_t|s_t)$ for $t = 0, \ldots, H-1$.

(d) What are the key benefits of using importance weighting to estimate the gradient of a target policy π_θ using data collected under a different policy π'? Why is it useful in practical settings?