1 Bandits and RL (20pt)

The following questions do not involve programming. You can use the markdown option for cells in Jupyter notebook to answer.

1. What is the difference between A/B testing and Multi-armed bandits?
2. What is the role of exploration in the Bandit problems?
3. What is the difference between the UCB and the Thompson sampling methods in terms of exploration?
4. How does the contextual setting differ from the non-contextual setting in terms of difficulty (be precise)?
5. Can bandit algorithms be used for contextual bandits setting? If so, what is the disadvantage?
6. What is the difference between a Markov Reward Process and a Markov Decision Process? Can Bellman Expectation Equation be applied to both?
7. What is the difference between supervised learning and reinforcement learning?
8. How are simulations used in a forward search? (i.e., in a simple Monte Carlo search)

2 Bandits (30pt)

Consider a 5-armed stochastic bandit problem with mean rewards of $(0.1, 0.1, 0.1, 0.1, 0.9)$. The arms are Bernoulli.

1. Write a function that responds with a stochastically generated reward given the arm index as an input. We will use it to test the performance of various algorithms next.
2. Write individual functions for epsilon-greedy, UCB1 (informally also referred to as UCB) and Thompson sampling (use Beta-Bernoulli conjugacy) from scratch.
3. For various choices of ϵ, show how epsilon-greedy performs in terms of cumulative expected regret and in terms of arm selection.

4. Plot multiple simulations of the performance of UCB1 algorithm.

5. Plot multiple simulations of the performance of Thompson sampling algorithm. Comment on which algorithm is better qualitatively.

3 Reinforcement Learning (30pt)

We will use the MIT licensed code available at `https://github.com/seungeunrho/minimalRL` to do sensitivity analysis of DQN for the cartpole environment from the gym package (see `https://gym.openai.com/`). You should clone it as needed.

1. Describe the state, actions, transitions and rewards for the cartpole environment using the gym package documentation.

2. Describe the Q-network used in 'dqn.py'. What are the layers and what are the outputs?

3. Run the default DQN configuration for cartpole in 'dqn.py' and plot the (25,50,75)-percentile reward performance curves over multiple simulations/runs.

4. Change the epsilon value (for exploration) to fixed values $\{0.01, 0.1\}$ and plot its impact on learning. Provide an interpretation of the trend observed.

5. Change the buffer_limit (of the experience replay buffer) to $\{5000, 10000, 25000\}$ and plot its impact on learning. Provide an interpretation of the trend observed.

6. Change gamma (for discounting) to $\{0.75, 0.9\}$ and discuss its impact on on learning. Provide an interpretation of the trend observed.