Reinforcement Learning: Policy Updates & Policy Gradients Quiz

Ethan B. Mehta
ethanbmehta@berkeley.edu

Sean Lin
seanlin2000@berkeley.edu

Jaiveer Singh
j.singh@berkeley.edu

1 MDP Questions

1.1 Q1

What are the 5 components of an MDP? Describe each one.

1.2 Q2

How is a Q-State different than a State? In what special cases would the Q-States and States be equivalent? *Hint: consider the size of the State and Action Spaces.*

1.3 Q3

Compute the size of the state space for the game of Monopoly with 2 players, 8 properties in total, 4 corner spaces (non-property locations). Each property can be owned by exactly 1 player or no player. Each player's token can be on one of the properties, or one of the corner spaces.

2 Offline Learning

2.1 Q4

Mortimer and Ellis are debating the formulation of the Bellman Equation.

1. Mortimer: The Bellman Equation is

$$V^*(s) = \max_{a \in actions} \sum_{s' \in states} T(s, a, s')[R(s, a, s') + \gamma \times V^*(s')]$$

   , because the optimal value of a state is the expected value of the Reward that an Agent starting from state s and behaving optimally afterwards will receive over its entire lifetime.
2. Ellis: I agree with your reasoning, but the Bellman Equation is

$$V^*(s) = \max_{a \in actions} Q(s, a)$$

Who is right? Why?

2.2 Q5

Anant, Jennifer, and Jitendra are sharing their favorite method of learning policies.

32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

1. Anant: My favorite method of learning policy is value iteration followed by policy extraction. I know that I always need my values to converge before I can figure out a policy.
2. Jennifer: My favorite method of learning policy is policy iteration. I don't like waiting for my values to converge because all I really need is a policy. If I do policy iteration, I can get the optimal policy without doing any value iteration!
3. Jitendra: My favorite method of learning policy is Q-Learning. I really like MDPs, so with Q-Learning I can learn the underlying MDP's transition functions and reward functions and then extract the optimal policy.

Why are they all wrong?

2.3 Q6

Why is the discount factor useful? What is the range of useful discount factors?

3 Online Learning

3.1 Q7

How does the Direct Evaluation algorithm incorporate information about the specific trajectory of States and Actions that the Agent takes? Is this a strength or a weakness of the Direct Evaluation method?

3.2 Q8

Considering Temporal Difference Learning, write the equations for each of the following. **Be sure to define all constants and Greek letters you reference.**

- Constructing a sample after a Transition has taken place
- Incorporating the sample into the estimated Values

3.3 Q9

Jethro tried to develop a new RL algorithm called J-Learning. The pseudocode of the J-Learning algorithm follows:

1. Initialize all Q-Values as 0
2. Repeat for the number of exploration episodes:
   (a) While the current State is not terminal:
      i. Use the current policy to determine the best Action
      ii. Transition to the new State using the calculated Action
      iii. Construct a sample using the incurred Reward
      iv. Update Q-Value using the new sample and learning rate alpha, with the same update equations as normal Q-Learning

Unfortunately, Jethro has made a critical error in this modification of Q-Learning. Explain the nature of Jethro's mistake, what step in the algorithm the error appears at, and what an effective solution might be.

4 Deep Reinforcement Learning

4.1 Q10

Alice and Bob both skimmed through the Note and have different understandings of what Policy Gradients are.

1. Alice: Policy Gradients involves computing the gradient of the Value function at each State, and then using Gradient Ascent to iteratively improve these estimates. The learned policy is Deterministic, implemented using an argmax operation over the Q-States.
2. Bob: Policy Gradients involves computing the gradient of the Policy parameters at each State, and then using Gradient Descent to iteratively improve these estimates. The learned policy is Stochastic, implemented using a Classification Neural Net.

Which parts of each student's explanations are correct? Formulate a concise explanation for the main idea behind Policy Gradients.

4.2 Q11

Briefly explain each part of A3C's full name: Asynchronous Advantage Actor-Critic.

4.3 Q12

What is one benefit of using OpenAI's Gym environment? Why is it useful for us to use the same environments as other RL researchers and engineers?