Reinforcement Learning: Policy Updates & Policy Gradients Quiz

Ethan B. Mehta
ethanbmehta@berkeley.edu

Sean Lin
seanlin2000@berkeley.edu

Jaiveer Singh
j.singh@berkeley.edu

0.1 Q1

What are States, Actions, and Rewards?

Answer:

- States: States represent the status of all the elements in the environment, encapsulating all the information that may change from time to time.
- Actions: Actions are the controllable choices an Agent can make when in a State, in order to initiate a Transition to a subsequent State.
- Rewards: Rewards are the positive or negative stimuli given to the Agent when it takes specific Actions. Typically, Reward is used to promote desirable Actions while discouraging poor Actions.

0.2 Q2

How does the Agent in Reinforcement Learning parallel the Control Law from EECS 16B? In what ways are they different?

Answer: A control law is a precise, fixed mathematical relationship between the state and the new input value. An Agent in RL functions on a similar mapping from State to Action, but generally has much more freedom in how it performs this mapping. As a result, Agents are much more versatile than typical control laws.

0.3 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.

Answer:

$(\text{\# of places for a player token})^{\text{\# of players}} \times \text{\# of property ownership states}^{\text{\# of properties}} = (8+4)^2 \times (2+1)^4 = 11664$

0.4 Q4

Anant, Jennifer, and Jitendra are talking about Policies. Here are snippets from their conversation:

1. Anant: A Policy is a way of representing how valuable each State is. It maps from State to estimated total Reward incurred from that State through the future.

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

2. Jennifer: Policies must always be deterministic; it doesn't make sense to have a Policy with any randomness (stochasticity).
3. Jitendra: If I'm dealing with a continuous State space, it makes sense to use a dictionary to store my Policy.

Why are each of their statements wrong?

Answer:

1. Anant's mistake is that the policy is a mapping from States to Actions, not States to Values.
2. Jennifer's mistake is that stochastic policies are actually a viable strategy, particularly for continuous state spaces.
3. Jitendra's mistake is that in a continuous State space, it is impossible to have a dictionary with discrete mappings from State to Action; this is why we use neural networks in Deep RL.

0.5 Q5

Concisely explain the idea of a Policy Gradient. What formulation of a Policy do we need to apply a Policy Gradient?

Answer: If we are able to parameterize our Policy, then we can take the gradient with respect to those parameters in order to progressively improve our Policy. This is a Policy Gradient approach.

0.6 Q6

Why does Deep RL typically use Gradient Ascent, instead of Gradient Descent? (Hint: what is RL seeking to optimize?)

Answer: RL is about maximizing Reward, while traditional ML is about minimizing loss. As a result, it makes more sense to perform Gradient Ascent in most Deep RL settings.

0.7 Q7

What is the purpose of the Classification Neural Net in Deep RL methods?

Answer: Since we often have a large, stochastic State space, the Classification Neural Net is employed to classify the State into one of several 'types' of States; based on the 'type', the Agent can choose the corresponding Action.

0.8 Q8

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

Answer:

1. Asynchronous: A3C uses multiple asynchronous worker Agents that work on their own, local copies of the model and environment. These Agents periodically update the global model once they find an improvement.
2. Advantage: A3C learns the Advantage instead of the Value function, exhibiting a preference to explore 'promising' areas of the State Space in which the actual Rewards were higher than the expected Rewards, instead of simply exploring the areas with high expected Rewards.
3. Actor-Critic: A3C uses an Actor-Critic system, in which the Actor attempts to choose the best Action given the current State, and in which the Critic evaluates the goodness of the Actor's choice.

0.9 Q9

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?

Answer: Using the OpenAI Gym environment provides a standardized set of problems to compare various RL algorithms. By ensuring that researchers across the world are using the same set of environments, the Gym makes it easy to identify which kinds of problems a new algorithm is better or worse at compared to the state-of-the-art.

0.10 Q10

Briefly describe the state space of the ‘CartPole-v0‘ environment. What is the objective?

Answer: The state space of the ‘CartPole-v0‘ environment includes the angle of the pole, the position of the cart, and the velocities of each. The goal is to keep the pole within a small deviation of vertical, while also keeping the cart within a small deviation of the center of the screen, for as long as possible.