Deep Reinforcement Learning (Sp25)

Instructor: Dr. Mohammad Hossein Rohban

Quiz Solutions - Lecture 25

Arranged by: Behnia Soleymani

Q1

Explain the core intuition behind the CQL (Conservative Q-Learning) regularizer:

$$\alpha \left(E_{s\sim D, a\sim\mu(a|s)}[Q(s,a)] - E_{(s,a)\sim D}[Q(s,a)] \right)$$

Specifically, what is the role of each expectation term, and what is the overall goal of this regularizer?

Answer:
$E_{(s,a)\sim D}[Q(s,a)]$: This term aims to push up (or accurately fit) the Q-values for state-action pairs that are actually present in the dataset. It anchors the Q-function to the observed data.
$E_{s\sim D, a\sim\mu(a|s)}[Q(s,a)]$: This term aims to push down the Q-values for actions sampled from a policy μ (which is often chosen to find actions that maximize Q, potentially OOD actions). It counteracts overestimation.
Overall Goal: The regularizer's goal is to learn a Q-function that is conservative by ensuring Q-values for in-dataset actions are accurate while simultaneously suppressing potentially overestimated Q-values for actions not well-supported by the data, thus preventing the policy from exploiting these erroneous high values.

Q2

In model-based offline RL, model exploitation is a significant challenge. MOPO (Model-Based Offline Policy Optimization) addresses this by:

A) Training an ensemble of models and only trusting predictions where all models agree.
B) Modifying the reward function used during planning/learning with the model to include a penalty proportional to the model's uncertainty about its predictions.
C) Applying a CQL-like penalty to Q-values derived from model-generated rollouts that deviate from the real data distribution.
D) Strictly limiting model rollouts to a single step to prevent error accumulation.

Correct Answer: B
Explanation: This accurately describes MOPO's uncertainty-based reward penalty. (a) is a technique for uncertainty estimation but not the core MOPO mechanism, (c) describes COMBO, (d) is an overly restrictive (and not primary) strategy.

Q3

Describe the difference in how CQL and policy constraint methods (e.g., KL-constrained policy optimization) attempt to address the problem of distributional shift in offline RL.

Answer: **Policy Constraint Methods:** These methods directly restrict the learned policy π to be close to the behavior policy π_β (or an estimate of it). This is often done by adding a constraint like $D_{KL}(\pi||\pi_\beta) \leq \epsilon$ to the policy optimization objective or by using weighted behavioral cloning (e.g., AWR). The primary focus is on ensuring the actions taken by the policy do not deviate too far from the data distribution.
CQL (Conservative Q-Learning): CQL focuses on regularizing the Q-function itself. It adds terms to the Q-learning objective that explicitly push down Q-values for actions likely to be OOD (found by μ) and push up/match Q-values for actions within the dataset. While the policy is then derived from this conservative Q-function, the direct intervention is on the value estimates, aiming to prevent overestimation for OOD actions, rather than directly constraining the policy's action space relative to π_β.

Q4

Which of the following BEST describes the primary reason the environment is considered non-stationary from a single agent's perspective in a Multi-Agent Reinforcement Learning (MARL) setting?

A) The underlying rules of the game change randomly over time.
B) The agent's own learning process causes its perception of the environment to shift.
C) Other adaptive agents are simultaneously learning and changing their policies.
D) The reward function for the agent is not clearly defined.

Correct Answer: C
Explanation: Other adaptive agents are simultaneously learning and changing their policies. As other agents learn and change their policies, the transition dynamics and reward structures effectively change over time, leading to non-stationarity. Option (a) is too general, (b) is an internal agent factor not the primary cause of environmental non-stationarity due to others, and (d) is a different problem.

Q5

Define a Nash Equilibrium. Why is it considered a stable outcome?

Answer: A Nash Equilibrium is a strategy profile (a set of strategies, one for each player) where no player can improve their own payoff by unilaterally changing their strategy (no incentive to deviate), assuming all other players stick to their strategies in that profile. It is considered stable because, once this state is reached, no individual agent has an incentive to deviate on their own; everyone is playing a best response to what others are doing.