DS 598 HW1

Write your name here

February 8, 2024

Instructions: Please submit a Jupyter notebook containing the code and the requested plots to the blackboard submission portal. You can make use of any open-sourced code as you wish, but you will be responsible for the correctness of the code you submit.

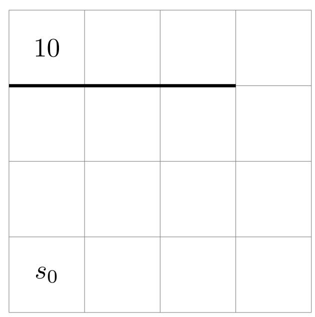

Figure 1: MDP.

Problem 1. Implement the MDP using the gym Env class. The game resets once the agent finds the treasure ($r = 10$). $\gamma = 0.9$. Each state has 4 actions which moves left, right, up and down, respectively. If the agent moves against a wall, it will remain in the current state.

Problem 2. Implement the DQN algorithm (with experience replay and target network) with ϵ-greedy exploration, i.e. the agent use the action $a_t = \arg\max_a Q_t(s_t)$ with probability $1 - \epsilon$ and $a_t = \text{Unif}(A)$ with probability ϵ. You are free to tune ϵ as well as other hyper-parameters in DQN as you wish. Use a two-layer MLP for the Q-network.

- Plot the learning curve averaging over 10 runs. The learning curve measures the performance of the policy $J(\pi_t)$ as a function of episode index t.
- Implement double DQN, and plot its learning curve in the same graph as above.

Problem 3. Implement the vanilla REINFORCE algorithm using a two-layer MLP network with softmax output layer. You can tune the mini-batch size and learning rate as you wish.

- Plot the learning curve averaging over 10 runs.
- Plot the learning curve using the V^* function as a baseline for variance reduction.
- Plot the empirical variance of policy gradient with and without baseline. Given a set of trajectories $\tau_{1:n}$ from a policy π_t, the empirical variance is

$$v_t = \frac{1}{n}\sum_{i=1}^{n} ||g_{t;i} - g_t||_2^2$$

where

$$\begin{aligned} g_{t;i} &= \sum_{h=0}^{\infty} \nabla_\theta \log \pi(a_{i;h}|s_{i;h})(R(s_{i;h}, a_{i;h}) - \text{baseline}(s_{i;h})) \\ g_t &= \frac{1}{n}\sum_{i=1}^{n} g_{t;i} \end{aligned}$$