

ME 418/518 – Data-Based Control

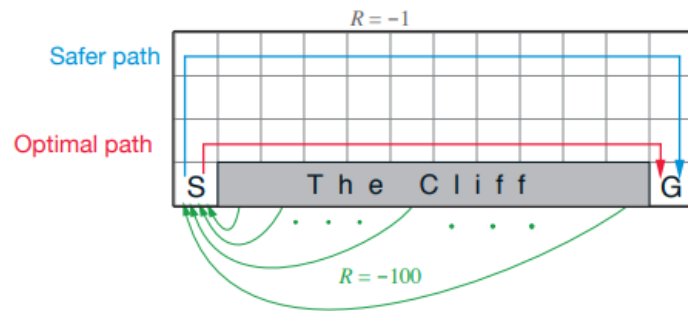
Problem Set 5

Deadline – December 18th, 2025

Employment of LLMs is strictly prohibited

Problem 1. Consider **Example 6.6: Cliff Walking** in our textbook. Reproduce and report your results. You also need to upload your code. The implementation should be written entirely by you, without using LLMs, online sources, RL libraries, or any built-in RL functions. Your solution should follow the pseudocode given in lecture notes (or textbook).

Example 6.6: Cliff Walking This gridworld example compares Sarsa and Q-learning, highlighting the difference between on-policy (Sarsa) and off-policy (Q-learning) methods. Consider the gridworld shown to the right. This is a standard undiscounted, episodic task, with start and goal states, and the usual actions causing movement up, down, right, and left. Reward is -1 on all transitions except those into the region marked “The Cliff.” Stepping into this region incurs a reward of -100 and sends the agent instantly back to the start.



The graph to the right shows the performance of the Sarsa and Q-learning methods with ϵ -greedy action selection, $\epsilon = 0.1$. After an initial transient, Q-learning learns values for the optimal policy, that which travels right along the edge of the cliff. Unfortunately, this results in its occasionally falling off the cliff because of the ϵ -greedy action selection. Sarsa, on the other hand, takes the action selection into account and learns the longer but safer path through the upper part of the grid. Although Q-learning actually learns the values of the optimal policy, its online performance is worse than that of Sarsa, which learns the roundabout policy. Of course, if ϵ were gradually reduced, then both methods would asymptotically converge to the optimal policy. ■

