Problem Session 3: MDP, Dynamic Programming and Model-free predicition
CS BDMA Machine Learning

Tom Dupuis

November 16, 2022

(1) Problem 1: MDP Design

You are in a Las Vegas casino! You have \$20 for this casino venture and will play until you lose it all or as soon as you double your money (i.e., increase your holding to at least \$40). You can choose to play two slot machines: 1) slot machine A costs \$10 to play and will return \$20 with probability 0.05 and \$0 otherwise; and 2) slot machine B costs \$20 to play and will return \$30 with probability 0.01 and \$0 otherwise. Until you are done, you will choose to play machine A or machine B in each turn. In the space below, provide an MDP that captures the above description. Describe the state space, action space, rewards and transition probabilities. Assume the discount factor $\gamma = 1$. Rewards should yield a higher reward when terminating with \$40 than when terminating with \$0. Also, the reward for terminating with \$40 should be the same regardless of how we got there (and equivalently for \$0).

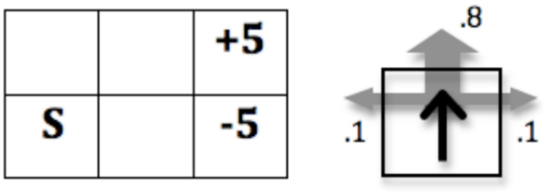
(2 Problem 2: Stochastic Optimal Policies

Given an optimal policy that is *stochastic* in an MDP, show that there is always another deterministic policy that has the same (optimal) value.

(3) Problem 3: Parallelizing Value Iteration During a single iteration of the Value Iteration algorithm, we typically iterate over the states in $\mathcal{S}$ in some order to update $V_t(s)$ to $V_{t+1}(s)$ for all states s. Is it possible to do this iterative process in parallel? Explain why or why not.

(4) Problem 4: Into the Unknown

Let us define a gridworld MDP, depicted in Figure 1. The states are grid squares, identified by their row and column number (row first). The agent always starts in state (1,1), marked with the letter S. There are two terminal goal states, (2,3) with reward +5 and (1,3) with reward -5. Rewards are 0 in non-terminal states. (The reward for a state is received as the agent moves into the state.) The transition function is such that the intended agent movement (Up, Down, Left, or Right) happens with probability .8. With probability .1 each, the agent ends up in one of the states perpendicular to the intended direction. If a collision with a wall happens, the agent stays in the same state.



		+5
S		-5

Figure 1: Left: Gridworld MDP, Right: Transition function

1. Define the optimal policy for this gridworld MDP.
   Since we know the transition function and the reward function, we can directly compute the optimal value function with value iteration. But, what if we don't know the transition and reward function?

2. Suppose the agent does not know the transition probabilities. What does the agent need to be able do (or have available) in order to learn the optimal policy?

3. The agent starts with the policy that always chooses to go right, and executes the following three trajectories: **1)** (1,1)–(1,2)–(1,3), **2)** (1,1)–(1,2)–(2,2)–(2,3), and **3)** (1,1)–(2,1)–(2,2)–(2,3). What are the First-Visit Monte Carlo estimates for states (1,1) and (2,2), given these trajectories? Suppose $\gamma = 1$.

4. Using a learning rate of $\alpha = 0.1$ and assuming initial values of 0 , what updates does the TD-learning agent make after trials 1 and 2, above? For this part, suppose $\gamma = 0.9$.

Exercises of Chapter 3, 4 and 5 of the RL Book (Sutton & Barto)