Lab Report 3

Eryka Liced Rimacuna Castillo Luis Catalán Salas

November 12, 2024

1 Questions

1.1 Question 1: Value Iteration

2. How are those functions related to the value iteration equation?

$$V_{k+1} = max \sum p(x'|x,a)[R(x,a,x') + \gamma V_k(x')]$$

This equation iteratively updates the value function $V(x)$ by finding the maximum expected reward achievable from each state x by taking the best action a and considering possible next states x'

ComputeActionFromValues = It computes $Q(x,a) = \sum p(x'|x,a)[R(x,a,x') + \gamma V_k(x')]$

computeQValueFromValues = In the value iteration equation, this corresponds to the maximization part $a^* = argmaxQ(x,a)$

1.2 Question 2: Bridge Crossing Analysis

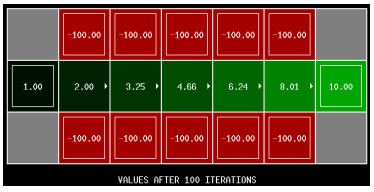


Figure 1: q2 gridworld

Reducing the noise to 0.01 minimizes the risk of the agent accidentally moving into the high-penalty states (the red tiles around the bridge) when trying to cross. With less noise, the agent is more confident that it can take its intended actions without being diverted, making the high-reward state (10.00) more attractive than the safer, low-reward state (1.00).

1.3 Question 3: Policies

python gridworld.py -a value -i 100 -g DiscountGrid –discount 0.1 –noise 0.0 –livingReward 0.0

1. discount=0.1, noise=0.0, livingReward=0.0

   The agent prefers the close exit (+1) and risks the cliff because the low discount (0.1) makes immediate rewards more valuable. The zero noise ensures a predictable path, allowing the agent to take the shortest route.

2. discount=0.1, noise=0.1, livingReward=0.0

   The agent still prefers the close exit (+1) but now avoids the cliff due to the slight noise (0.1). This small noise introduces risk in taking direct paths, so the agent chooses a safer route away from the cliff.

3. discount=0.1, noise=0.0, livingReward=0.9

   The agent prefers the distant exit (+10) and risks the cliff because the high living reward (0.9) makes longer paths more rewarding. Low discount and zero noise push the agent toward the high-reward exit via the shortest risky route.

4. discount=0.1, noise=0.1, livingReward=0.9

   The agent prefers the distant exit (+10) and avoids the cliff. Here, the combination of high

living reward and slight noise encourages a longer, safer route to the distant exit.

5. discount=0.0, noise=0.0, livingReward=10.1

   The agent avoids both exits and the cliff because the very high living reward (10) discourages any termination of the episode. The agent prefers to keep moving indefinitely to maximize ongoing rewards, avoiding exits altogether.

1.4 Question 4 and 5: Q-Learning and Epsilon Greedy

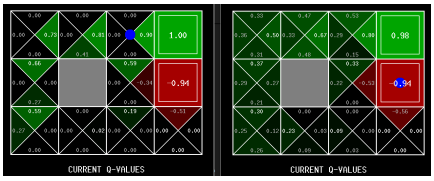


Figure 2: q5 gridworld

With an epsilon value of 0.1 (left image), the agent mostly exploits the best-known actions, occasionally exploring. This leads to more stable and converging Q-values. With epsilon at 0.9 (right image), the agent explores randomly most of the time, resulting in more varied Q-values and slower convergence toward the optimal policy. This high exploration causes the agent to make choices that deviate significantly from the most rewarding paths.

1.5 Question 6: Bridge Crossing Revisited

Figure 3: q6 gridworld

With epsilon = 1 (full exploration (top image)), the agent explores randomly and doesn't focus on optimizing any particular path. This results in an average return of -62.96, indicating that it often encounters high-penalty states due to the frequent exploration of suboptimal paths.

With epsilon = 0 (pure exploitation), the agent strictly follows its learned policy based on priormo knowledge. This leads to a much better average return of -4.37 because it avoids unnecessary risky moves, resulting in a more stable and optimal path selection.

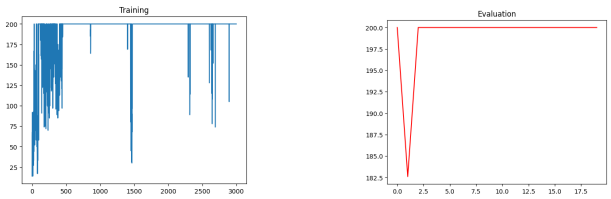
Mathematically, we the probability of selecting the correct move each time is $\frac{1}{4}$. Therefore, the probability of choosing the correct sequence of five actions to reach the goal state is:

$$\left(\frac{1}{4}\right)^5 = \frac{1}{1024} \approx 0.00098 \text{ (or } 0.098\%).$$

This extremely low probability shows that, with purely random exploration, the agent has less than a 0.1% chance of finding the optimal path in any single episode. Given only 50 episodes, the probability of discovering and reinforcing this path enough to achieve consistent success is negligible.

1.6 Question 9: Policy search of a Cart-Pole system

1.6.1 CartPole-v0



(a) Training (b) Evaluation

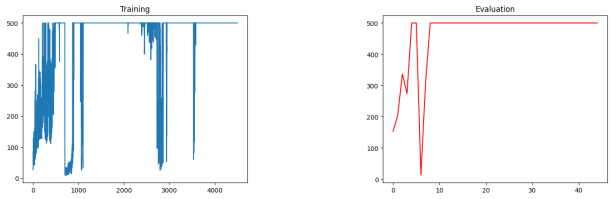
Figure 4: Graphics CartPole-v0.

Training: The agent initially shows high variability in rewards, with frequent dips, indicating exploration. By around 1500 episodes, it consistently achieves high rewards, though occasional drops still occur.

Evaluation: After initial instability, the agent consistently achieves the maximum reward of 200, show-

ing that it has learned a stable policy for balancing the pole.

1.6.2 CartPole-v1



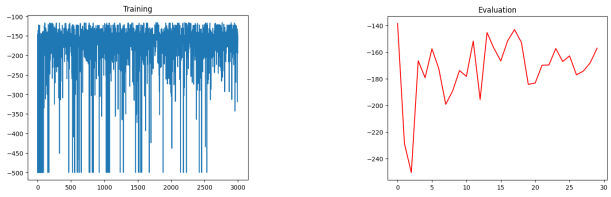
(a) Training (b) Evaluation

Figure 5: Graphics CartPole-v1.

Training: The rewards initially fluctuate as the agent learns, with some episodes reaching the maximum reward of 500. By around 1000 episodes, the agent consistently performs well, although there are occasional drops even later in training.

Evaluation: After an initial phase with variable rewards, the agent quickly stabilizes, achieving the maximum reward of 500 in all subsequent evaluations, indicating a reliable and effective policy.

1.6.3 Acrobot-v1



(a) Training (b) Evaluation

Figure 6: Graphics Acrobot-v1.

Training: The rewards fluctuate widely, showing some episodes close to -100 but frequent drops near -500, indicating instability in learning. The agent has difficulty consistently reaching the goal, as seen from the variability.

Evaluation: Although there's gradual improvement over time, the evaluation scores mostly stay around -140 to -200, suggesting only moderate learning progress. The agent occasionally performs well but struggles to maintain high performance consistently.

This demonstrates that while basic policy gradient methods can handle simple control tasks, they may not generalize well to more challenging reinforcement learning environments without further improvements to the model complexity or learning approach.