Name: yousef alaa saad

id: 20210477

Name: martina mazzouz sabry

id: 20210309

FrozenLake RL Algorithms Report

1. Algorithm Overviews

Value Iteration is a dynamic programming algorithm used to compute the optimal policy for Markov Decision Processes (MDPs). It works by iteratively updating the value of each state using the Bellman optimality equation until the value function converges. Once convergence is achieved, the optimal policy is extracted by selecting the action that maximizes the expected value at each state.
Key points:
- It assumes full knowledge of the environment's transition probabilities.
- It is guaranteed to converge to the optimal policy if the MDP is finite.

Q-Learning is a model-free reinforcement learning algorithm. It learns the optimal action-value function (Q-function) by interacting with the environment and updating its estimates based on the Temporal Difference (TD) learning approach.

Key points:
- Does not require a model of the environment.
- Uses an ε-greedy strategy to balance exploration and exploitation.
- Gradually improves the Q-values over episodes until a near-optimal policy is learned.

---

2. Implementation Approach

1- Environment Setup

- Environment: FrozenLake-v1 (4x4 map, deterministic is_slippery=False).
- Rewards: +1 for reaching the goal, 0 otherwise.

2- Value Iteration Implementation

- Initialize state-value function V to zeros.
- At each iteration:
  - Update the value of each state based on the best action value.
  - Track the maximum change (delta) to monitor convergence.
- Stop when the maximum change falls below a small threshold (theta=1e-6).
- Extract the policy by choosing the action with the highest expected value at each state.

3- Q-Learning Implementation

- Initialize a Q-table with zeros.
- For each episode:

- Use an ε-greedy strategy for action selection.
- Update the Q-value using the TD learning rule.

---

3. Results and Performance Comparison

Algorithm	Success Rate %	Convergence Speed
Value Iteration	100	Fast (15 iterations)
Q-Learning	100	Slower (after ~2000 episodes)

4. Discussion

- Value Iteration converged much faster since it had access to the full environment dynamics.
- Q-Learning required significantly more interactions with the environment to achieve comparable performance because it had to learn through trial and error.
- Both algorithms eventually achieved optimal policies leading to nearly 100% success rates in evaluation.
- Visualization of the learned policies confirmed that both approaches found the shortest and safest path to the goal.

---

5. Conclusion

- Value Iteration is ideal when the environment model is available and small enough to allow tabular methods.
- Q-Learning is more flexible and realistic for larger or unknown environments, despite its slower learning speed.
- Both methods successfully solved the FrozenLake environment when tuned properly.