CSCI 397: Value Iteration for Frozen Lake

Due Date: Friday October 20th @ 11:59 PM

Information

Implement the value iteration algorithm to find the optimal policy for navigating the FrozenLake environment in gymnasium.

The FrozenLake environment models a grid world where an agent must navigate from a starting position to a goal position while avoiding holes. The environment is both slippery and stochastic, meaning that the agent does not always move in the direction it intends to.

Tasks

Environment Exploration:

> Instantiate the FrozenLake environment with is_slippery=True.
>
> Explore the environment by taking random actions and observe the rewards and next states.
>
> Understand the reward structure and the total number of states and actions.

Implement Value Iteration:

> Initialize a value table with zeros.
>
> For a given number of iterations or until convergence:
>
>> For each state, perform a "look-ahead" for each possible action and compute the expected value.
>>
>> Update the value of the state based on the expected maximum value.
>
> Note: Ensure you consider the environment's transition probabilities due to its stochastic nature.

Extract Policy:

> For each state, choose the action that has the maximum expected value.

Evaluation:

> Use the extracted policy to evaluate its performance over multiple episodes in the environment.
>
> Compute the average reward and report the percentage of episodes in which the agent successfully reached the goal.

Experiment:

Compare the performance of the agent when is_slippery=True vs. is_slippery=False.

How does the stochasticity of the environment affect the value iteration process and the resulting policy?

Turn in

Python code implementing value iteration for FrozenLake.

A brief report discussing your findings, including:

The final value table.

The extracted policy in a readable format.

Evaluation results.

The number of iterations until convergence