**CSC 580 AI II| HW 2-1** | **Name: Om Prakash Gunja** | **Student ID: 2131025**

**Part 1: Fixed Policies**

**Objective**

Evaluate the performance of predefined fixed policies in the Cliff Walking environment by measuring: **Step Count**, **Near-Falls** and **Cumulative Reward**

**Methodology**

The run() function was modified to:

* Record **Step Count**, **Near-Falls**, and **Cumulative Reward** for each run.
* Track two types of "near-falls" cause I didn’t understand how to estimate that properly:
  1. **Beside Cliff**: Counted when the agent was in cells adjacent to the cliff (e.g., positions 24-36 and 47) and could potentially fall off.
  2. **Near fall**: Counted when the agent attempted a move into the cliff but avoided falling due to the stochastic nature of the environment.
* Return all metrics for further analysis, ensuring flexibility in comparing policies.

**So in the output graph the Ratio of Around cliff matches with the expected Ratio of near falls as instructed in the assignment**

These adjustments allowed for detailed performance comparisons between policies based on multiple metrics and interpretations of "near-falls."

A two-step process was implemented to evaluate policies and visualize results:

1. **Procedure Function**: Executes the policy across episodes and records metrics.
2. **Output Function**: Computes means, standard deviations, and visualizes results via histograms for steps, rewards, near-falls, and beside-cliff ratios.

**Results for Seed = 17**

|  |  |  |
| --- | --- | --- |
| **Metric** | **Mean** | **Standard Deviation** |
| **Reward** | -20748.47 | 20842.68 |
| **Steps** | 3127.46 | 3045.67 |
| **Near-Falls** | 5.91e-5 | 2.07e-4 |
| **Beside Cliff Ratio** | 0.4136 | 0.0499 |

**A graph of a number of points

AI-generated content may be incorrect.Steps and Rewards**: High variability indicates inconsistent navigation, with frequent penalties leading to negative rewards. **Near-Falls** are rare but suggest limited exploration of risky areas. The **Beside Cliff Ratio** highlights that significant time near the cliff contributes to penalties, emphasizing the need for improved risk-aware navigation.

**Lets do this for 3 more fixed ( random ) policies :**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Metric** | **Mean (Seed 19)** | **Std Dev (Seed 19)** | **Mean (Seed 21)** | **Std Dev (Seed 21)** | **Mean (Seed 23)** | **Std Dev (Seed 23)** |
| **Reward** | -25970.70 | 27633.32 | -3166.11 | 3125.73 | -43920.51 | 38895.26 |
| **Steps** | 2706.69 | 2809.72 | 639.63 | 581.71 | 4552.17 | 4004.78 |
| **Near-Falls** | 0.1464 | 0.0198 | 0.0658 | 0.0208 | 0.1231 | 0.0221 |
| **Beside Cliff** | 0.4631 | 0.0591 | 0.2135 | 0.0577 | 0.3786 | 0.0659 |

**A green and red graph

AI-generated content may be incorrect.**

**A graph of a red and green column

AI-generated content may be incorrect.**

**A graph of a graph of a graph

AI-generated content may be incorrect.**

**Rewards**: Seed 21 had the least negative reward (-3166.11), indicating superior performance. Seed 23 showed the worst outcomes (-43920.51).

**Steps**: Seed 21 required the fewest steps (639.63), showcasing task efficiency. Seed 23 had the highest steps (4552.17).

**Safety**: Seed 21 was safest, with the lowest near-falls (0.0658) and minimal beside-cliff movements (0.2135). Seed 19 exhibited the riskiest behavior.

**Variability**: Seed 23 showed high variability across metrics, suggesting policy instability.

**Part 2: Policy Iteration**

**To implement MDP I used textbook reference from instructions**

**A screenshot of a computer code

AI-generated content may be incorrect.**For this part I have made 4 functions

* 1. getInitialPolicy(seed) : Initializes the **Value Function (V)** and **Policy**

A computer code with text

AI-generated content may be incorrect.

* 1. policy\_evaluation(env, V, policy, gamma, theta) : Iteratively updates the **Value Function (V)** for a given policy until convergence.

A computer screen shot of a program code

AI-generated content may be incorrect.

* 1. policy\_improvement(env, V, policy, gamma): Improves the given policy using the updated Value Function.
  2. A screen shot of a computer code

     AI-generated content may be incorrect.policy\_iteration(seed): Runs **Policy Iteration** to compute the optimal policy and value function.

Instead of just running on one random policy I wanted to see if the seed value actually had anything?

I ran it through [17, 19, 21, 23, 25]

A white background with numbers and symbols

AI-generated content may be incorrect.Analysis for Seed 17:

A number of numbers and symbols

AI-generated content may be incorrect.

A number of numbers on a white background

AI-generated content may be incorrect.A number of numbers and symbols

AI-generated content may be incorrect.

Optimal policy is logical for the **Cliff Walking** problem.

* **Third and Fourth Row:** Mostly **stay (0)**, likely avoiding risky moves due to high penalties. The behavior aligns with expected reinforcement learning outcomes.

1. Performance Comparison

* **Optimal Rewards**: The average reward for seed 17 is **-65.41**, with a standard deviation of **27.38**. This indicates moderate variability in performance across episodes.
* **Steps to Convergence**: Policy iteration converged in **17 iterations**, which is relatively fast.
* **Stability**: The low standard deviation for near falls (**0.0299**) and the standard deviation of beside-cliff (**0.1079**) suggest stable behavior.

2. Effectiveness of the Optimal Policy

* **Improvement Over Fixed Policies**: The optimal policy shows a significant improvement compared to the initial random fixed policy:
  + The initial value function was uniformly small, indicating no meaningful understanding of the environment, while the optimal value function has distinct patterns.
  + The optimal policy directs the agent towards the goal efficiently, avoiding unnecessary exploration near the cliff.
* **Converged Value Function Patterns**:
  + The values are highest near the goal (state 47 at the bottom-right corner).
  + A sharp drop in value indicates the cliff area (bottom row, columns 1–10), emphasizing the penalty associated with falling.
* **Grid Behavior**:
  + The optimal policy avoids the cliff effectively, as shown by the directed movement along the top and middle rows. And expected behavior in the grid (4x12) aligns with reinforcement learning principles, as the agent takes the shortest and safest route to the goal.

3. Convergence Insights

* **Fastest Convergence**: The seed converged in just **17 iterations**, demonstrating the effectiveness of the initial policy and value estimates.
* **Stability**:
  + Minimal near falls indicate that the agent successfully learned to avoid dangerous zones.
  + Moderate variability in rewards and steps suggests occasional exploration around risky areas but with controlled outcomes.
* **Unexpected Behavior**:
  + There are no significant signs of instability or unexpected behavior. However, the reward variability suggests that some episodes may still encounter near-cliff scenarios, albeit rarely.

A graph of a red and green column

AI-generated content may be incorrect.

compared to random policy

|  |  |  |
| --- | --- | --- |
| **Metric** | **Optimal Policy** | **Random Policy** |
| **Average Reward** | -67.28 | -20748.47 |
| **Std Reward** | 27.50 | 20842.68 |
| **Average Steps** | 67.28 | 3127.46 |
| **Std Steps** | 27.50 | 3045.67 |
| **Average Near-Falls** | 0.045 | 5.91e-5 |
| **Std Near-Falls** | 0.030 | 2.07e-4 |
| **Beside Cliff Ratio** | 0.370 | 0.4136 |
| **Std Beside Cliff** | 0.100 | 0.0499 |

The **optimal policy dramatically outperforms the random policy** in both efficiency and stability. The random policy struggles to navigate effectively, leading to **longer episodes, unstable performance, and significantly lower rewards**.

Performance Trends Across Seeds

| **Metric** | **Seed 17** | **Seed 19** | **Seed 21** | **Seed 23** | **Seed 25** |
| --- | --- | --- | --- | --- | --- |
| **Average Reward** | -67.28 | -67.19 | -68.54 | -66.6 | -67.27 |
| **Std Reward** | 27.50 | 25.51 | 25.85 | 22.79 | 29.01 |
| **Average Steps** | 67.28 | 67.19 | 68.54 | 66.6 | 67.27 |
| **Std Steps** | 27.50 | 25.51 | 25.85 | 22.79 | 29.01 |
| **Average Near-Falls** | 0.045 | 0.044 | 0.044 | 0.045 | 0.049 |
| **Std Near-Falls** | 0.030 | 0.028 | 0.031 | 0.034 | 0.032 |
| **Beside Cliff Ratio** | 0.370 | 0.372 | 0.350 | 0.380 | 0.363 |
| **Std Beside Cliff** | 0.100 | 0.107 | 0.099 | 0.130 | 0.097 |

The results across seeds showed consistent patterns with minimal differences in performance metrics such as rewards, steps, and safety ratios.

I also tried with various intial V(s) to see if that has any effect ..

Comparison of Performance with Different Initial V(s) Values (Seed 17)

|  |  |  |  |
| --- | --- | --- | --- |
| **Metric** | **V(s) = 0.1** | **V(s) = 0.5** | **V(s) = 1.0** |
| **Average Reward** | -63.94 | -68.64 | -61.45 |
| **Std Reward** | 24.43 | 30.24 | 21.32 |
| **Average Steps** | 63.94 | 68.64 | 61.45 |
| **Std Steps** | 24.43 | 30.24 | 21.32 |
| **Average Near-Falls** | 0.0468 | 0.0459 | 0.0471 |
| **Std Near-Falls** | 0.0345 | 0.0320 | 0.0289 |
| **Beside Cliff Ratio** | 0.3739 | 0.3674 | 0.3537 |
| **Std Beside Cliff** | 0.1153 | 0.1078 | 0.1106 |

All setups demonstrated similar safety levels, with slight variations in near-falls and the beside-cliff ratio.

Lets see if gamma has any effect ..

Comparison of Performance with Different Gamma (γ) Values (Seed 17)

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Metric** | **γ = 0.1** | **γ = 0.5** | **γ = 0.8** | **γ = 0.9** | **γ = 0.99** |
| **Converged Iterations** | 8 | 24 | 17 | 11 | 4 |
| **Average Reward** | -72.75 | -66.42 | -66.38 | -65.96 | -65.41 |
| **Std Reward** | 32.07 | 30.04 | 21.38 | 25.08 | 22.55 |
| **Average Steps** | 72.75 | 66.42 | 66.38 | 65.96 | 65.41 |
| **Std Steps** | 32.07 | 30.04 | 21.38 | 25.08 | 22.55 |
| **Average Near-Falls** | 0.0437 | 0.0507 | 0.0440 | 0.0428 | 0.0481 |
| **Std Near-Falls** | 0.0288 | 0.0351 | 0.0318 | 0.0273 | 0.0321 |
| **Beside Cliff Ratio** | 0.3232 | 0.3452 | 0.3688 | 0.3570 | 0.3345 |
| **Std Beside Cliff** | 0.1179 | 0.1144 | 0.1076 | 0.1047 | 0.1011 |

**Higher γ values (0.9, 0.99) lead to faster convergence and better performance**.

**Lower γ values (0.1, 0.5) result in longer episodes and suboptimal rewards**.

**trade-off**: Higher γ encourages **long-term reward optimization** but may take slightly riskier routes.

Summary

* This assignment was quite interesting and helped me understand how reinforcement learning works in practical settings.
* Initially, the policy was random and not useful, but after running the policy iteration, the optimal policy started making sense.
* One surprising thing was in the **third row**, where the agent preferred **left (0)** or no movement instead of taking other actions.
* The hardest part of this task was **interpreting the results** and checking if the learned policy makes sense.
* Overall, this exercise was a great way to understand **policy iteration** and how reinforcement learning agents optimize decisions over time.
* I did use multiple llms to format the document