Reinforcement Learning Lab Exercise 3

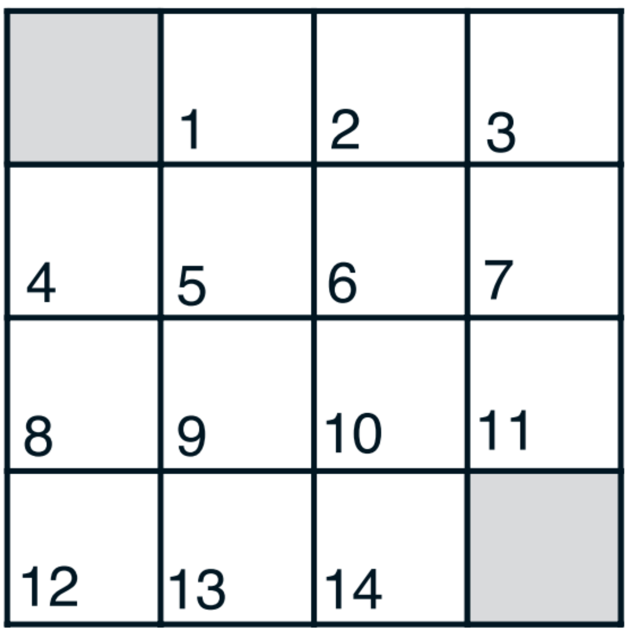
Jinay Vora
Student ID: 202201473

Problem Set 2

Description

From [3], study example 4.1 and write a Python code to find vπ(s).

Solution



Python Code

```
import numpy as np

def in_bounds(row, col):
    return 0 <= row < 4 and 0 <= col < 4

def state_to_rowcol(s):
    s0 = s - 1
    return divmod(s0, 4)
```

```python
def rowcol_to_state(row, col):
    return row * 4 + col + 1

def next_state(s, action):
    if s in [1, 16]:
        return s

    row, col = state_to_rowcol(s)

    if action == 'U':
        new_row, new_col = row - 1, col
    elif action == 'D':
        new_row, new_col = row + 1, col
    elif action == 'L':
        new_row, new_col = row, col - 1
    elif action == 'R':
        new_row, new_col = row, col + 1
    else:
        raise ValueError("Unknown action!")

    if in_bounds(new_row, new_col):
        return rowcol_to_state(new_row, new_col)
    else:
        return s

def iterative_policy_evaluation(theta=1e-8, max_iterations=100000):
    V = np.zeros(17)
    gamma = 1.0
    actions = ['U', 'D', 'L', 'R']

    for _ in range(max_iterations):
        delta = 0
        newV = V.copy()

        for s in range(1, 17):
            if s in [1, 16]:
                continue

            v_new = 0.0
            for a in actions:
                s_next = next_state(s, a)
                reward = -1
                v_new += 0.25 * (reward + gamma * V[s_next])
            newV[s] = v_new

        delta = np.max(np.abs(newV - V))
```

```python
        V = newV
        if delta < theta:
            break

    return V

if __name__ == "__main__":
    V_final = iterative_policy_evaluation()
    print("Value function with terminal states at (0,0) and (3,3):\n")

    for row in range(4):
        row_str = []
        for col in range(4):
            s = row * 4 + col + 1
            row_str.append(f"{V_final[s]:6.2f}")
        print(" ".join(row_str))
```

Solution

```
Value function with terminal s


  0.00 -14.00 -20.00 -22.00
-14.00 -18.00 -20.00 -20.00
-20.00 -20.00 -18.00 -14.00
-22.00 -20.00 -14.00   0.00
```

Problem Set 3

Description

From [3], solve exercises 4.1 and 4.2. You can choose to solve it using a Python code.

Solution

Exercise 4.1

We know that,

$$q_\pi(s,a) = \sum_{s'} P(s'|s,a)\left[R(s,a,s') + \gamma v_\pi(s')\right]$$

But for deterministic actions,

$$q_\pi(s,a) = R(s,a,s') + \gamma v_\pi(s')$$

Using the above two equations, we get:

(a) $q_\pi(11, \text{down}) = -1 + 1 \cdot (0) = -1$

(b) $q_\pi(7, \text{down}) = -1 + 1 \cdot (-14) = -15$

Exercise 4.2

Part 1: Original Dynamics

The value function for state 15 is:

$$v_\pi(15) = -1 + \frac{1}{4}\left(v_\pi(12) + v_\pi(13) + v_\pi(14) + v_\pi(15)\right)$$

Part 2: Changed Dynamics

If action down from state 13 leads to 15, the Bellman equations for $v_\pi(13)$ and $v_\pi(15)$ are:

$$v_\pi(13) = -1 + \frac{1}{4}(v_\pi(12) + v_\pi(13) + v_\pi(14) + v_\pi(15))$$

$$v_\pi(15) = -1 + \frac{1}{4}(v_\pi(12) + v_\pi(13) + v_\pi(14) + v_\pi(15))$$

Rearrange both equations:

$$3v_\pi(15) = -4 + v_\pi(12) + v_\pi(13) + v_\pi(14)$$

$$3v_\pi(13) = -4 + v_\pi(12) + v_\pi(14) + v_\pi(15)$$

Substitute $v_\pi(15)$ into the equation for $v_\pi(13)$:

$$v_\pi(13) = \frac{-4 + v_\pi(12) + v_\pi(14) + \frac{-4+v_\pi(12)+v_\pi(13)+v_\pi(14)}{3}}{3}$$

Simplifying gives:

$$v_\pi(13) = 0.5v_\pi(12) + 0.5v_\pi(14) - 2$$

By symmetry, $v_\pi(15)$ is:

$$v_\pi(15) = 0.5v_\pi(12) + 0.5v_\pi(14) - 2 = 0.5(-22) + 0.5(-14) - 2 = -20$$

Problem Set 4

Description

From [3], solve exercise 4.3.

Solution

Analogous to Equation (4.3):

$$q_\pi(s,a) = \mathbb{E}_\pi\left[R_{t+1} + \gamma q_\pi(S_{t+1}, A_{t+1}) \mid S_t = s, A_t = a\right]$$

Analogous to Equation (4.4):

$$q_\pi(s,a) = \sum_{s',r} p(s', r \mid s, a)\left[r + \gamma \sum_{a'} \pi(a'|s') q_\pi(s', a')\right]$$

Analogous to Equation (4.5):

The iterative update rule for successive approximations of q_π is:

$$q_{k+1}(s,a) = \mathbb{E}_\pi\left[R_{t+1} + \gamma q_k(S_{t+1}, A_{t+1}) \mid S_t = s, A_t = a\right]$$