Q-Learning agent report

For the third question, we were asked to implement a Q-Learning based agent. We implemented train(), extractPolicy() and a epsilon-greedy policy function.

The below output is tested against a user

Choose location to put your O based on the following scheme.

0|1|2

3|4|5

6|7|8

Your move: 0

Playing move: O(0,0)

|O| | |

| | | |

| | | |

Playing move: X(0,1)

|O|X| |

| | | |

| | | |

Choose location to put your O based on the following scheme.

0|1|2

3|4|5

6|7|8

Your move: 7

Playing move: O(2,1)

|O|X| |

| | | |

| |O| |

Playing move: X(0,2)

|O|X|X|

| | | |

| |O| |

Choose location to put your O based on the following scheme.

0|1|2

3|4|5

6|7|8

Your move: 8

Playing move: O(2,2)

|O|X|X|

| | | |

| |O|O|

Playing move: X(1,0)

|O|X|X|

|X| | |

| |O|O|

Choose location to put your O based on the following scheme.

0|1|2

3|4|5

6|7|8

Your move: 6

Playing move: O(2,0)

|O|X|X|

|X| | |

|O|O|O|

O won!

The below output is tested against the provided test cases

Against Defensive agent:

X won!

Wins: 37 Losses: 12 Draws: 1