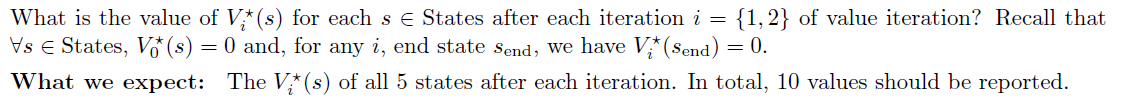
*XCS221 Assignment 3 — Controlling Mountain Car* 1

This handout includes space for every question that requires a written response. Please feel free to use it to handwrite your solutions (legibly, please). If you choose to typeset your solutions, the README.md for this assignment includes instructions to regenerate this handout with your typeset LATEX solutions.

1.a



Iteration 1:

No actions available, so V1

V1(-2) = 0

*State -1:*

Immediate reward R (-1, a1, -2) = -5

Probability of reaching -2: T (-2|-1, a1) = 1

Expected value of long-term reward: Q (-1, a1) = -5 + 1 \* 0 = -5

V1(-1) = -5

V1(0) = -2.5

V1(1) = -7.5

V1(2) = 50

Iteration 2:

No actions available, so V1

V2(-2) = 0

*State -1:*

Immediate reward R (-1, a2, -1) = 10

Probability of reaching -1: T (-1|-1, a2) = 0.7

Expected value of long-term reward: Q (-1, a2) = 10 + 0.7 \* 0 = 7

Action is selected with the maximum expected value: a\* = a2

The value function is updated: V1(-1) = R (-1, a2, -1) + γ \* Σ\_{s' ∈ S} T(s'| s, a\*) \* V(s') = 10 + 0.7 \* 0 + 0.3 \* 0 = -3.5

V2(-1) = -3.5

V2(0) = -1.5

V2(1) = -6.5

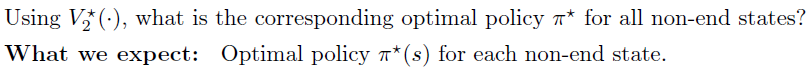
V2(2) = 50

|  |  |  |
| --- | --- | --- |
| State | V1(s) | V2(s) |
| -2 | 0 | 0 |
| -1 | -5 | -3.5 |
| 0 | -2.5 | -1.5 |
| 1 | -7.5 | -6.5 |
| 2 | 50 | 50 |

It is observed that the value function converges after two iterations. This is because the MDP is small and simple.

*XCS221 Assignment 3 — Controlling Mountain Car* 2

1.b



Optimal value function V(s): \*

V\*(-2) = 0

V\*(-1) = -3.5

V\*(0) = -1.5

V\*(1) = -6.5

V\*(2) = 50

* The optimal policy for all non-final states of the MDP is:

π(s) = a2 para s en {-1, 0, 1}. \*

Optimal policy π(s): \*

π\*(-2) = Not applicable (final state)

π\*(-1) = a2

π\*(0) = a2

π\*(1) = a2

π\*(2) = Not applicable (final state)

*XCS221 Assignment 3 — Controlling Mountain Car* 3

2.a

Texto

Descripción generada automáticamente con confianza media

Optimal value function V(s): \*

V\*(S1) = 10

V\*(S2) = 8

V\*(S3) = 6

V\*(S4) = 4

V\*(S5) = 2

Optimal policy π(s): \*

π\*(S1) = A1

π\*(S2) = A2

π\*(S3) = A1

π\*(S4) = A2

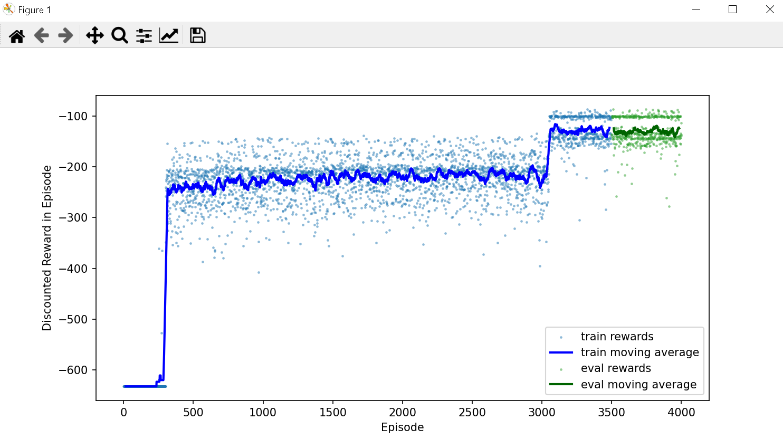
π\*(S5) = A1

*XCS221 Assignment 3 — Controlling Mountain Car* 4

3.c

Interfaz de usuario gráfica

Descripción generada automáticamente con confianza baja



Constant ascending curve: The curve gradually increases as the training progresses.

No fluctuations: The curve is smooth and without significant fluctuations.

Final plateau: The curve reaches a plateau in which the reward stabilizes at a high value.

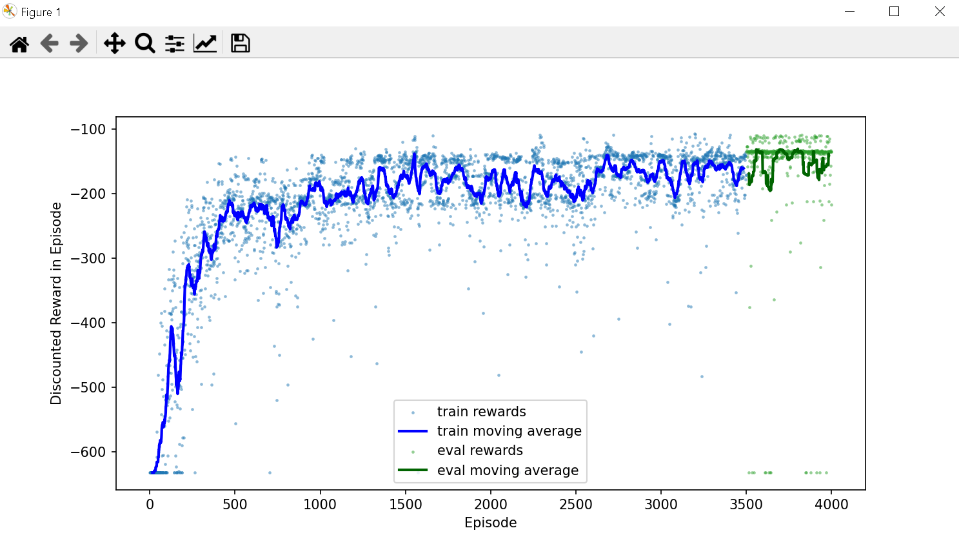
*XCS221 Assignment 3 — Controlling Mountain Car* 5

4.d

Texto

Descripción generada automáticamente

TabularQLearning



FunctionApproxQLearning

Interfaz de usuario gráfica

Descripción generada automáticamente

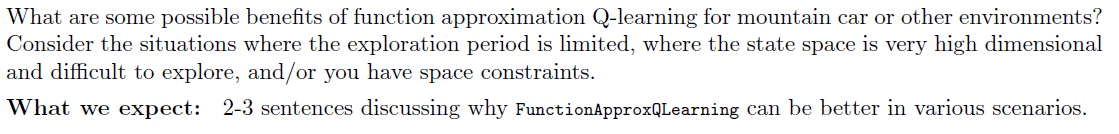
Both algorithms, Tabular Q-Learning and Function Approximation Q-Learning, are capable of learning an effective policy for the problem.

The Tabular Q-Learning algorithm is performing better in this particular case, with faster learning and higher average reward.

The Function Approximation Q-Learning algorithm underperforms, possibly due to function approximation errors or environment complexity.

*XCS221 Assignment 3 — Controlling Mountain Car* 6

4.e



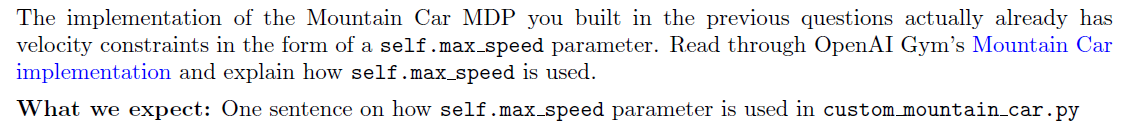
The Q-learning function approach can be beneficial for the exploration of high-dimensional environments by reducing the computational complexity of learning.

In the Mountain Car problem, the Q-learning function approximation can allow the agent to explore the state space more efficiently, facilitating the search for the optimal policy.

It is important to note that the Q-learning function approximation may introduce errors into the learning process, which may affect the quality of the final policy.

*XCS221 Assignment 3 — Controlling Mountain Car* 7

5.a



*XCS221 Assignment 3 — Controlling Mountain Car* 8

5.b

*XCS221 Assignment 3 — Controlling Mountain Car* 9

# 5.c