RL using Temporal Difference

By Sefunmi Ashiru

Environment: OpenAI gym FrozenLake8x8-v0

(S: starting point, safe)
(F: frozen surface, safe)
(H: hole, fall to your doom)
(G: goal, where the frisbee is located)
(: agent)

State 1	State k
SFFF	SFFF
FHFH	FHFH
FFFH	FFFH
HFFG	HFFG

Q-Learning

Tabular method

Problems in which the state and actions spaces are small enough for approximate value functions to be represented as arrays and tables.

	Action 1 (up)	Action 2 (left)	Action 3 (right)	Action 4 (down)
State 1	Q(s1,a1)	Q(s1,a2)	Q(s1,a3)	Q(s1,a4)
State 2	Q(s2,a1)	Q(s2,a2)	Q(s2,a3)	Q(s2,a4)
State 3	Q(s3,a1)	Q(s3,a2)	Q(s3,a3)	Q(s3,a4)
|				
State k	Q(sk,a1)	Q(sk,a2)	Q(sk,a3)	Q(sk,a4)

Pros

- Doesn’t require the transition probability matrix like in Dynamic Programing.
- Updates Q/”state action pair” values incrementally. Don't need to wait till episode ends.

Cons

- Q-Learning can behave poorly in some stochastic environments.

Algorithm

Episodes = 200000
Learning Rate:
α = .648
Discounted Rewards:
γ = .9

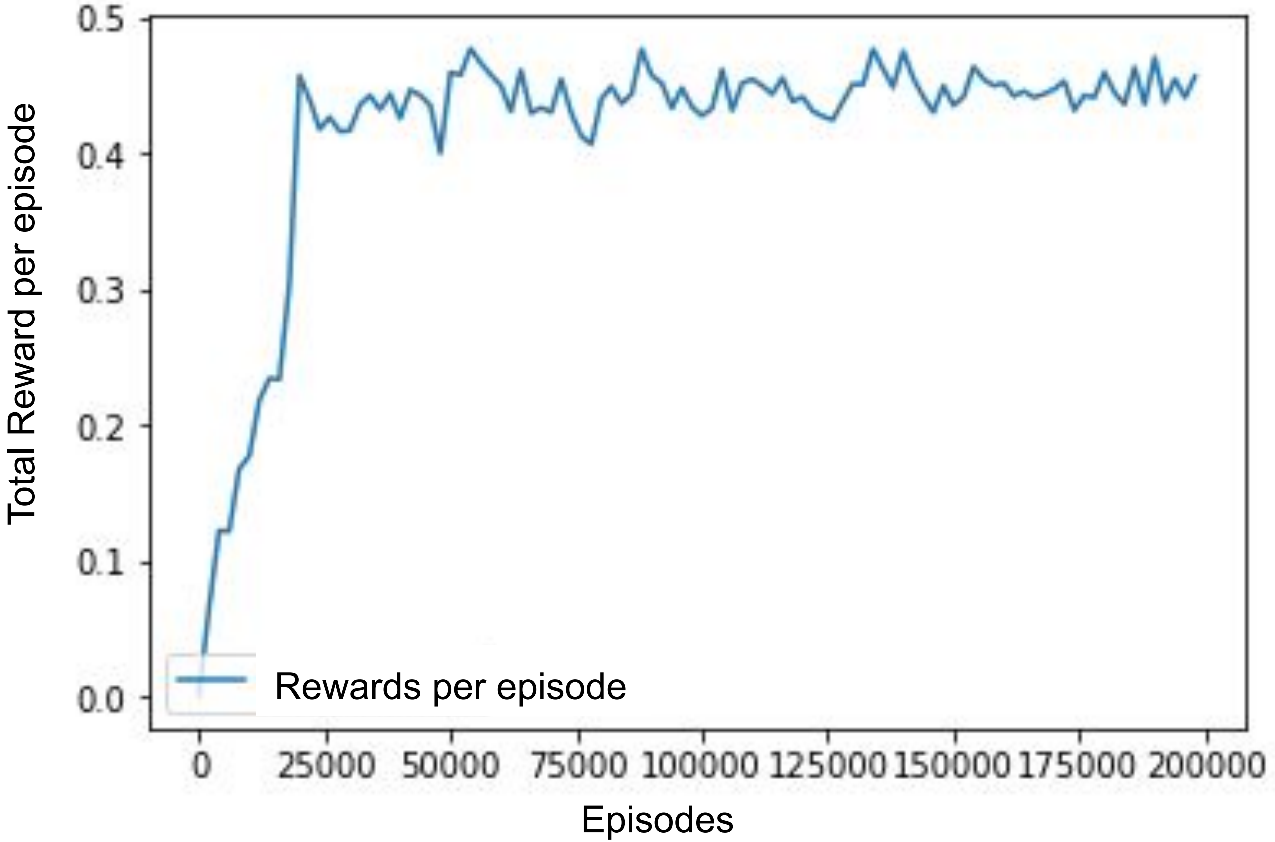
Q-learning (off-policy TD control) for estimating $\pi \approx \pi_*$

Algorithm parameters: step size $\alpha \in (0, 1]$, small $\varepsilon > 0$
Initialize $Q(s, a)$, for all $s \in \mathcal{S}^+, a \in \mathcal{A}(s)$, arbitrarily except that $Q(terminal, \cdot) = 0$

Loop for each episode:
  Initialize S
  Loop for each step of episode:
    Choose A from S using policy derived from Q (e.g., ε-greedy)
    Take action A, observe R, S'
    $Q(S, A) \leftarrow Q(S, A) + \alpha\big[R + \gamma \max_a Q(S', a) - Q(S, A)\big]$
    $S \leftarrow S'$
  until S is terminal

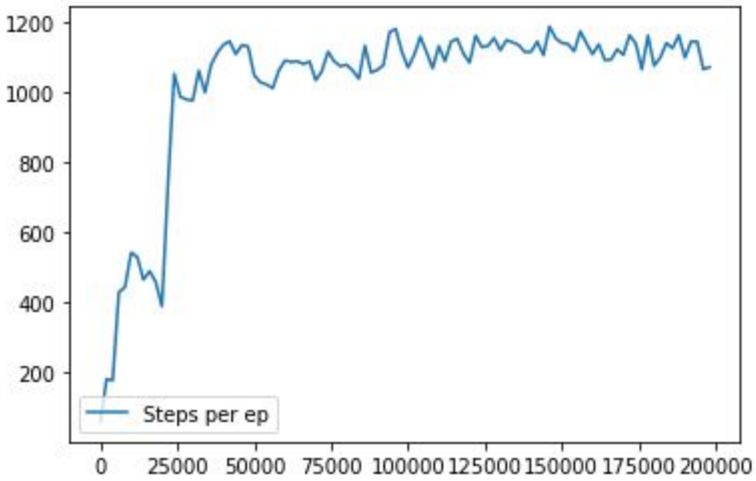
[1] (Salloum, 2018)

Rewards per episode



Bibliography

1. Salloum, Z. (2018, December 16). Summary of Tabular Methods in Reinforcement Learning [Web log post]. Retrieved 2020, from https://towardsdatascience.com/summary-of-tabular-methods-in-reinforcement-learning-39d653e904af#:~:text=Introduction,represented%20as%20arrays%20and%20tables.



1200
1000
800
600
400
200
Steps per ep
0
25000
50000
75000
100000
125000
150000
175000
200000