Samuel Barr - (18038795)

SARSA Algorithm

State-Action-Reward-State-Action (SARSA) is model-free Reinforcement Learning algorithm used to find the optimum policy for a Markov Decision Process (MDP). It was first proposed in “Online Q-Learning using Connectionist Systems" by (Rummery and Niranjan, 1994) It is a member of the Temporal-Difference (TD) class of algorithms similar to Q-Learning and is an on-policy algorithm, meaning that it requires an initial policy to iterate on.

SARSA often uses ϵ-greedy (epsilon-greedy) as its initial policy as it provides a balance between exploration and exploitation. According to (Sutton and Barto,1998), SARSA converges with probability 1 to an optimal policy π^* and action-value function $Q(a, s)$ after all state-action pairs are visited an infinite number of times.

Algorithm

Firstly, for all state and action pairs (s, a), initialise the action-value function $Q(s, a)$ with any number. From the initial state, the next action is chosen by the policy π. For a given state s, $Q(s, a)$ is be updated using equation 1. Figure 1 shows the states, rewards and actions used in the equation.

Where:

- r is the immediate reward after taking action a in state s.
- Υ is the discount factor where $[0 \leq \Upsilon \leq 1]$
- $s_{t+1}$is the next state.
- a_{t+1} is the next action according to the policy.
- α is the learning rate $0 \leq \alpha \leq 1$

$$Q(s_t, a_t) = Q(s_t, a_t) + \alpha[r_{t+1} + \Upsilon Q(s_{t+1}, a_{t+1}) - Q(s_t, a_t)\,] \quad (1)$$

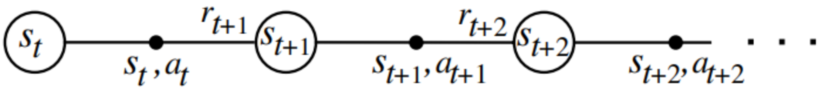
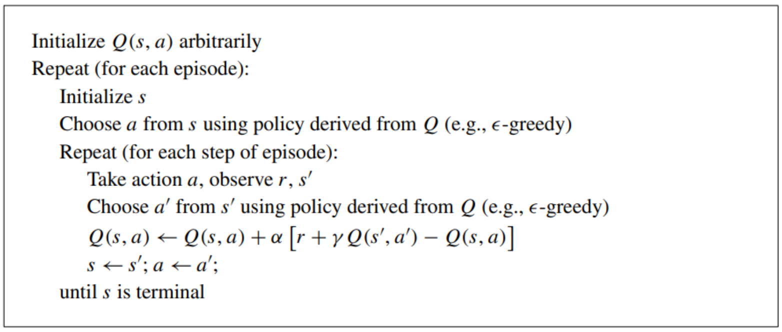


Figure 1 – State transitions (Sutton and Barto,1998 pg. 145)

The current state s is then shifted to state s_{t+1}. This process is repeated for every state transition until a termination state is reached or all Q-values converge. As the algorithm continues, $Q(s, a)$ converges to $Q^*(s, a)$ and the optimal policy π^* is found.

The pseudo-code for SARSA is shown in Figure 2 provided by (Sutton and Barto,1998, pg 143).



```
Initialize $Q(s, a)$ arbitrarily
Repeat (for each episode):
    Initialize $s$
    Choose $a$ from $s$ using policy derived from $Q$ (e.g., $\epsilon$-greedy)
    Repeat (for each step of episode):
        Take action $a$, observe $r$, $s'$
        Choose $a'$ from $s'$ using policy derived from $Q$ (e.g., $\epsilon$-greedy)
        $Q(s, a) \leftarrow Q(s, a) + \alpha \left[ r + \gamma Q(s', a') - Q(s, a) \right]$
        $s \leftarrow s'$; $a \leftarrow a'$;
    until $s$ is terminal
```

Figure 2 – SARSA pseudo-code

References

Rummery, G. A. and Niranjan, M. (1994) *Online Q-Learning using Connectionist Systems.* : Cambridge University Engineering Department

Sutton, R.S. and Barto, A.G. (1998) *Reinforcement Learning: An Introduction*. 2nd ed. : Mit Press.