Hands on reinforcement learning

Winter in data science

Mayank Gupta

January 29, 2023

1 Reinforcement Learning

Reinforcement learning is an area of Machine Learning where an agent has the basic purpose of interacting with the environment and earning a reward for each action performed. The agent has the sole purpose of finding a policy to optimally maximise the reward obtained.

2 Elements involved in Reinforcement learning

2.1 Policy

A policy decides how does the agent behave depending on the present state and the action performed by the agent.

2.2 Reward Signal

A reward signal is the input received by the agent on performing the action. The agent's sole purpose is to maximise the reward obtained from the reward signal.

2.3 Model

A model mimics the behaviour of the environment and predicts how will the environment behave. A model can predict the next state and action by the current state and the action taken before.

2.4 Exploiting

At any time step, if the agent selects the the action whose reward is the highest, it is called a greedy action and the agent is then exploiting its current knowledge.

2.5 Exploring

If a non greedy action is selected, it is called exploring as it improves the estimate of the nongreedy action's value.

3 Action value

The actual value of any action a is denoted by $q(a)$. The estimated value on the tth step is $Q_t(a)$. If action a has been chosen $N_t(a)$ times before t, and the rewards at these steps are $R_1, R_2, ..., R_{N_t(a)}$, then:

$$Q_t(a) = \frac{R_1 + R_2 + ... + R_{N_t(a)}}{N_t(a)} \tag{1}$$

By law of large numbers, as $N_t(a) \to \infty$, $Q_t(a) \to q(a)$

3.1 Greedy method

The greedy action method selects the action which gives the highest reward at that particular time.

$$A_t = \arg\max_a Q_t(a) \tag{2}$$

3.2 ϵ greedy method

In this method, the most greedy action is chosen most of the time, but at few time instants a action is chosen randomnly with probability ϵ.

4 Assignment on Bandits

An assignment which included implementing the algorithms for the ϵ greedy and the upper confidence bound algorithm. This was done on a boiler plate code provided which included functions on bandits and thompson sampling

algorithm. The curve for the average reward in Thompson and ϵ greedy algorithms are:

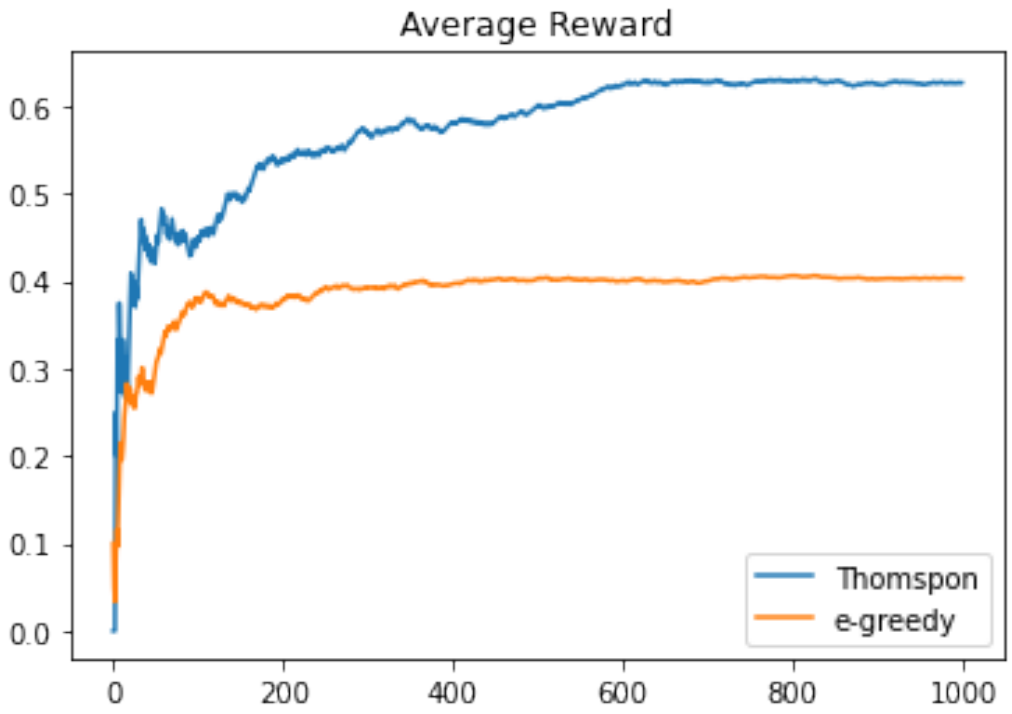

Figure 1: Curve for the average reward in Thompson and ϵ greedy algorithms

The regret vs horizon graph is:

Figure 2: Curve for the regret vs horizon for Thompson and ϵ greedy algorithms

5 Bellman Optimality operator

Let $S = \{s_1, s_2, ..., s_n\}$. A function F is defined as $F : S \to R$. The Bellman optimality operator, $B^* : R^n \to R^n$ for a Markov decision process defined as (S, A, T, R, γ) as:

$$(B^*(F))(s) \equiv \max_{a \in A} \sum_{s' \in S} T(s, a, s')\{R(s, a, s') + \gamma F(s')\} \tag{3}$$

Here, B^* is a contraction mapping.

5.1 Banach's fixed point theorem

Banach's fixed point theorem implies that there is a unique fixed point for B^*. Let this fixed point be $V^* : S \to R$. Therefore, $B^*(V^*) = V^*$.

$$V^*(s) = \max_{a \in A} \sum_{s' \in S} T(s, a, s')\{R(s, a, s') + \gamma V^*(s')\} \tag{4}$$

6 Dynamic programming

Three algorithms to compute V^* are Value iteration, linear programming, and policy iteration.

6.1 Value iteration

This is an iterative approach to compute V^*.

An assignment to implement the dynamic programming algorithm value iteration from scratch was completed. The psuedo code for the algorithm is: V_0 is a bounded arbitary n length vector.

$t \leftarrow 0$

while $V_t \neq V_{t-1}$ for all states For all states $s \in S$
$V_{t+1}(s) = \max_{a \in A} \sum_{s' \in S} T(s, a, s')\{R(s, a, s') + \gamma V_t(s')\}$ $t \leftarrow t + 1$

6.2 Policy iteration

Policy iteration involves the selection of a random policy π initially. As long as π is improvable, π will be improved to a better policy.

Psuedo code for policy iteration is: While π is improvable: $\pi' \leftarrow$ Improve policy $\pi \leftarrow \pi'$

7 Conclusion

Reinforcement learning is a goal directed approach for decision making. It doesn't focus on intermediate steps for achieving the goal. For example in chess the agent only focusses on winning the match and not on the intermediate steps to improve points by capturing more pieces. Reinforcement learning uses a formal framework defining the interaction between a learning agent and its environment in terms of states, actions, and rewards. It learns

from direct interaction with the environment. It doesn't rely on any other complete model of the environment. It can be used both at a high and low level. The agent uses its experience to improve its performance over time.