Notes from
Reinforcement Learning - An Introduction
Chapter 2

Jacek Plocharczyk

October 25, 2018

This is the short summary of each chapter from *Reinforcement Learning - An Introduction* by Richard S. Sutton and Andrew G. Barto. Please note that this is an unauthorized material. I will put my effort to provide the best quality I can but please bare in mind that some error and misunderstandings can occur.

Abstract

This is the summary of 2nd chapter from *Reinforcement Learning - An Introduction* by Richard S. Sutton and Andrew G. Barto. This notes are focused mostly on theory and equations.

1 A k-armed Bandit Problem

Expected value of arbitrary action a is described as q_*:

$$q_*(a) = \mathbb{E}[R_t | A_t = a] \tag{1}$$

Main goal of reinforcement learning is to find optimal ration between exploration and exploitation.

1.1 Action-value Methods

Estimation of q_* of action a in time t is denoted by Q_t:

$$Q_t(a) = \frac{\sum_{i=1}^{t-1} R_i \cdot \mathbb{1}_{A_i = a}}{\sum_{i=1}^{t-1} \mathbb{1}_{A_i = a}} \tag{2}$$

For number of iteration $n \to \infty$ the estimated action-value function $Q \to q_*$.

Greedy action is the action with the highest estimated reward:

$$A_t = \underset{a}{arg\,max} Q_t(a) \tag{3}$$

1.2 Incremental Implementation

We can simplify description of estimated action value function of single action:

$$Q_{n+1} = Q_n + \frac{1}{n}[R_n - Q_n] \tag{4}$$

Algorithm 1 Simple bandit algorithm

procedure
    for $a = 1$ to k **do**
        $Q(a) = 0$
        $N(a) = 0$
    while forever **do**
        $A = \begin{cases} \underset{a}{arg\,max} Q(a) & \text{with probability } 1 - \epsilon \\ \text{random action} & \text{with probability } \epsilon \end{cases}$
        $R = bandit(A)$
        $N(A) = N(A) + 1$
        $Q(A) = Q(A) + \frac{1}{N(A)}[R - Q(A)]$

1.3 Tracking Nonstationary Problem

For nonstationary problems we can use constant step-size parameter α in range $(0, 1]$:

$$Q_{n+1} = Q_n + \alpha[R_n - Q_n] \tag{5}$$

1.4 Optimistic Initial Values

To boost initial convergence of action-value function we can add some constant to $Q_1(a)$ which cause better exploration at the beginning.

1.5 Upper-Confidence-Bound Action Selection

When we need to include uncertainty about our estimations we can use method called *Upper-Confidence-Bound Action Selection* which choose action based on following rule:

$$A_t = \underset{a}{arg\,max}\left[Q_t(a) + c\sqrt{\frac{\ln t}{N_t(a)}}\right] \tag{6}$$

where $c > 0$ controls the degree of exploration.

1.6 Gradient Bandit Algorithm

Using numerical *preference* instead action-values.

$$\Pr\{A_t = a\} = \frac{e^{H_t(a)}}{\sum_{b=1}^{k} e^{H_t(b)}} = \pi_t(a) \tag{7}$$

where $\pi_t(a)$ is probability of taking action a in time-step t and $H_t(a)$ is a preference of taking action a in time-step t:

$$\begin{aligned} H_{t+1}(A_t) &= H_t(A_t) + \alpha(R_t - \bar{R}_t)(1 - \pi_t(A_t)), && \text{and} \\ H_{t+1}(a) &= H_t(a) + \alpha(R_t - \bar{R}_t)\pi_t(a), && \text{for all } a \neq A_t \end{aligned} \tag{8}$$