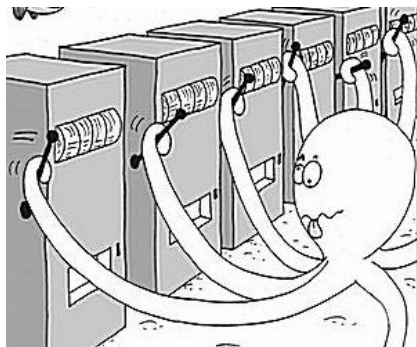


1 Multi-Armed Bandits: UCB and Hoeffding's inequality

In the last lecture, we began talking about multi-armed bandits. In the multi-armed bandits setting, we consider a decision-maker who is given K options to choose from. We refer to these options as “arms”. Associated with each arm is a probability distribution over rewards. Initially, this distribution is unknown to the decision-maker. The decision-maker chooses an arm, usually referred to as pulling an arm, and receives a reward sampled from the corresponding reward distribution. This process is repeated over and over again.



The problem we want to solve is to decide which arm to pull at each time step. One possible algorithm for deciding which arm to pull is the Upper Confidence Bound (UCB) algorithm presented in lecture. If we assume that the reward of each arm is bounded (e.g. the slot machine returns between \$0 and \$100), then proved in lecture that the UCB algorithm has bounded regret over time.

One thing that we didn't do in lecture is actually derive where UCB algorithm came from. In this discussion, we will derive the UCB algorithm using the Hoeffding bound. As we did in lecture, we will assume that the reward of each arm is bounded.

1. We first set up the framework of a multi-armed bandit problem. Suppose you have a set of K “arms”, $\mathcal{A} = \{1, 2, \dots, K\}$. Each arm $a \in \mathcal{A}$ has its own reward distribution $X_a \sim \mathbb{P}_a$ with mean $\mu_a = \mathbb{E}[X_a]$. Define the number of times arm a has been pulled up to and including time t as $T_a(t)$. In these problems we do not know μ_a but we would like to efficiently find the arm with the maximum mean by creating an algorithm that balances *exploration* of the arms with *exploitation* of the best possible arm. The efficiency of the algorithm is measured by a theoretical quantity known as regret, which measures how well the algorithm performs in expectation against an ‘oracle’ that knows the means of all the arms and always pulls the arm with highest mean.

We will now derive the upper confidence bound that yields the UCB algorithm we saw in the last lecture (up to a constant factor).

The general formula for constructing an upper confidence bound for the true mean μ_a of an “arm” a , given $T_a(t)$ samples $X_a^{(1)}, \dots, X_a^{(T_a(t))}$, is to find a value of $C_a(T_a(t), \delta)$ such that:

$$P(\mu_a < \hat{\mu}_{a,T_a(t)} + C_a(T_a(t), \delta)) > 1 - \delta \quad (1)$$

where $\hat{\mu}_{a,T_a(t)}$ is the sample mean of the reward from arm a , given by $\hat{\mu}_{a,T_a(t)} = \frac{1}{T_a(t)} \sum_{i=1}^{T_a(t)} X_a^{(i)}$. In words, Equation (1) says that with probability at least $1 - \delta$, the true mean μ_a is less than the estimated mean $\hat{\mu}_{a,T_a(t)}$ plus an upper confidence bound $C_a(T_a(t), \delta)$.

(a) Suppose that you know that the reward of any arm is between 0 and 1. That is:

$$X_a \in [0, 1]$$

Construct an upper confidence bound $C_a(T_a(t), \delta)$ for the mean of arm a , after observing t samples from arm a .

Solution: Hoeffding’s inequality bounds the difference between $\hat{\mu}_{a,T_a(t)}$ and μ_a . Hoeffding’s inequality works in two directions:

$$P(\hat{\mu}_{a,T_a(t)} - \mu_a \geq \epsilon) \leq e^{-2T_a(t)\epsilon^2} \quad (2)$$

$$P(\hat{\mu}_{a,T_a(t)} - \mu_a \leq -\epsilon) \leq e^{-2T_a(t)\epsilon^2} \quad (3)$$

The goal is to find the unknown quantity $C_a(T_a(t), \delta)$ from Equation (1) in terms of δ and $T_a(t)$. We now rearrange Equation (1) so that we can apply Hoeffding’s inequality.

$$\begin{aligned} P(\mu_a < \hat{\mu}_{a,T_a(t)} + C_a(T_a(t), \delta)) &> 1 - \delta \\ P(\hat{\mu}_{a,T_a(t)} - \mu_a > -C_a(T_a(t), \delta)) &> 1 - \delta \end{aligned}$$

Subtracting both sides from 1,

$$\begin{aligned} 1 - P(\hat{\mu}_{a,T_a(t)} - \mu_a > -C_a(T_a(t), \delta)) &\leq \delta \\ \implies P(\hat{\mu}_{a,T_a(t)} - \mu_a \leq -C_a(T_a(t), \delta)) &\leq \delta \end{aligned}$$

This looks a lot more like the form of Hoeffding’s inequality in Equation (3). Remember that the goal here is to compute the unknown quantity $C_a(t, \delta)$ in terms of δ and t . If we replace ϵ in Hoeffding’s inequality (Equation (3)) with $C_a(T_a(t), \delta)$, we have

$$P(\hat{\mu}_{a,T_a(t)} - \mu_a \leq -C_a(T_a(t), \delta)) \leq e^{-2T_a(t)C_a(T_a(t), \delta)^2} \quad (4)$$

To compute the value of $C_a(T_a(t), \delta)$ that achieves the desired probability bound of δ from Equation (1), we can set the right hand side of Equation (4) to δ :

$$e^{-2T_a(t)C_a(T_a(t), \delta)^2} = \delta$$

$$\implies C_a(T_a(t), \delta) = \sqrt{\frac{-\log \delta}{2T_a(t)}}$$

Plugging this value of $C_a(T_a(t), \delta)$ back into Equation (1), we have

$$P\left(\mu_a < \hat{\mu}_{a, T_a(t)} + \sqrt{\frac{-\log \delta}{2T_a(t)}}\right) > 1 - \delta.$$

- (b) Suppose we set $\delta = \frac{1}{t^3}$. This controls the probability that the true mean μ_a is greater than our upper confidence bound $C_a(T_a(t), \delta)$ on the estimated mean $\hat{\mu}_{a, T_a(t)}$. What rule does the UCB algorithm use to choose an arm A_t at each iteration t ?

Solution: The UCB algorithm optimistically chooses the arm with the highest estimated mean plus upper confidence bound. At time step t , we have access to mean and upper confidence bound estimates from up to time $t - 1$. Therefore, at time step t , the UCB algorithm chooses arm A_t as follows:

$$A_t = \operatorname{argmax}_{a \in \mathcal{A}} \hat{\mu}_{a, T_a(t-1)} + C_a(T_a(t-1), \delta)$$

Plugging in the upper confidence bound $C_a(T_a(t-1), \delta)$ from part (a), we have

$$A_t = \operatorname{argmax}_{a \in \mathcal{A}} \hat{\mu}_{a, T_a(t-1)} + \sqrt{\frac{-\log \delta}{2T_a(t-1)}}$$

Substituting in our choice of small $\delta = \frac{1}{t^3}$,

$$A_t = \operatorname{argmax}_{a \in \mathcal{A}} \hat{\mu}_{a, T_a(t-1)} + \sqrt{\frac{-\log \frac{1}{t^3}}{2T_a(t-1)}}$$

$$\implies A_t = \operatorname{argmax}_{a \in \mathcal{A}} \hat{\mu}_{a, T_a(t-1)} + \sqrt{\frac{3 \log t}{2T_a(t-1)}}$$

This final result matches the UCB algorithm presented in lecture.