TP2 : Bandits

Urtzi AYESTA and Matthieu JONCKHEERE

You may work in pairs, but please submit an individual report answering all the questions, and always explaining your answers. Please do not write only "numbers", explain briefly every step, and provide the code you used. In your report, indicate the name of the person you have worked with.

Submit your report in a single PDF. We will set a deadline during the TP.

You can use a python notebook provided by us to help you start or code directly from scratch.

We look here at bandits where the rewards are only 1 (with probability p) or 0 (Bernoulli random variables). Do simulations with a bandit with 10 arms.

Define a simulation where playing bandit k returns a 1 reward with probability p_k. By default, take the number of episodes to be $T = 1000$, and repeat every experiment 2000 times. In every experiment, select randomly a vector $p \in (0, 1)^{10}$.

1. Basic Algorithms

Consider the following policies: ϵ-greedy rule, optimistic greedy, and the UCB rule.
Questions:

1. Plot the corresponding mean reward; the mean cumulative reward and the percentage of times the best arm was elected as time goes by. Play with the parameters (ϵ and c, and T) and study their effects.
2. With ϵ-greedy, what is the asymptotic probability of taking the optimal action?
3. Which ϵ is better for a relatively small of T? and for large T
4. Do you observe some spikes in the plot of average rewards? if yes, please provide an explanation.
5. Plot the corresponding mean reward; the mean cumulative reward and the percentage of times the best arm was elected as time goes by. Interpret.
6. What are your conclusions in terms of methods? Give some intuition.

2. Gradient method

Implement the gradient bandit algorithm. Consider the same experiment setting as above, but with a reward of 4 instead of 1.

Consider both cases, with a baseline $\bar{R} = 0$, and $\bar{R} = 4 \times \max_k p_k$. Consider different values of the alpha parameter $\alpha_n = \alpha_0/n$.

As performance measure, consider as above the mean reward; the mean cumulative reward and the percentage of times the best arm was elected as time goes by.
Questions:

1. Which method does work better, with or without baseline?
2. How does the performance of the gradient method compares to the other?

3. Parameter study by Learning curve

In this section we want to compare in a systematic way the performance of all policies. To do this, we need to consider the performance as a function of the parameters of each method.

For each of 4 algorithms considered above, plot the average reward over 1000 steps as a function of its own parameters, see Figure .

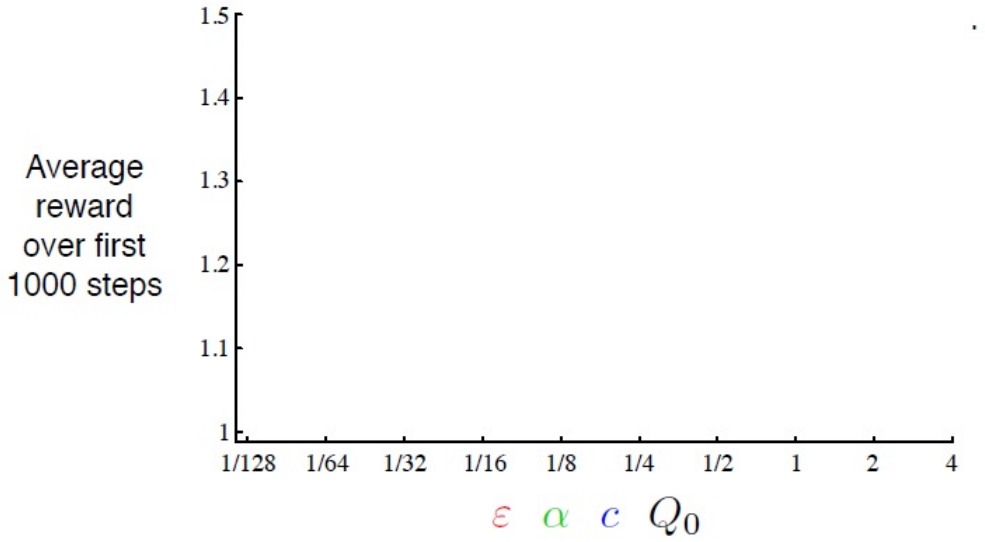

Figure 1: Example of parameter study. For the four algorithms, the respective values are varied by factors of two and presented on a log scale. ϵ for ϵ-greedy, α for gradient, c for UCB, and Q_0 for optimistic-greedy.

Questions:

1. What is the best algorithm for this example ?
2. Can you comment on the shape of the curves? what is the optimal parameter in each of the algorithms?