CS6046: Multi-Armed Bandits
Assignment 2

Course Instructor : Arun Rajkumar.

Release Date : April-06, 2021

Submission Date: On or before 5 PM on May-10,2021

SCORING: There are 4 question in this assignment which contributes to 10 points towards your final grade. Each question carries equal points.

WHAT SHOULD YOU SUBMIT? You should submit a zip file titled 'Solutions_ roll-number.zip' Your assignment will NOT be graded if it does not contain all of the following:

- A PDF file which includes explanations regarding each of the solution as required in the question. Title this file as 'Report.pdf'
- Source code for all the programs that you write for the assignment clearly named.

CODE LIBRARY: You are expected to code all algorithms from scratch. You cannot use standard inbuilt libraries for the algorithms. You are free to use inbuilt libraries for plots. You can code using either Python or Matlab or C.

GUIDELINES: Keep the below points in mind before submission.

- Plagiarism of any kind is unacceptable. These include copying text or code from any online sources. These will lead to disciplinary actions according to institute guidelines.
- Any graph that you plot is unacceptable for grading unless it labels the x-axis and y-axis clearly.
- Don't be vague in your explanations. The clearer your answer is, the more chance it will be scored higher.

LATE SUBMISSION POLICY You are expected to submit your assignment on or before the deadline to avoid any penalty. Late submission incurs a penalty equal to the number of days your submission is late by.

Q1: Randomizing the Leader: Consider the problem of online learning on the simplex Δ_d where d = 1000; At round t, you predict p_t and receive a vector z_t and suffer a loss of $p_t^T z_t$. Assume the adversary picks the vector z_t as the t-th row in the dataset *Dataset_Z*. Implement FTRL with quadratic and entropic regularization for this problem and plot the regret over time.

Now, consider the following algorithm which first picks and fixes a random 1000 dimensional vector R sampled uniformly from $[0, 1/\eta]^d$ and uses the following rule for prediction

$$p_{t+1} = \arg\min_{p \in \Delta_d} \sum_{i=1}^{t} \left(p^T(z_i + R)\right)$$

How would you choose η for this problem? For the value chosen, plot the regret bound for this algorithm as well. How does the regret bound compare with the previous two algorithms for this problem?

Q2: Mean Wise Variance Foolish?

The goal of this question is to study the variance of the EXP3 algorithm that was discussed in class. Let the number of arms be 10. At each iteration, generate a 10 dimensional loss vector where each component is chosen iid from a Beta distribution with both parameters equal to 5 for all arms except the 10th arm. For the 10th arm, let the parameters be $\alpha = 5, \beta = 10$. Implement the algorithm and run it for 1000 times for $10,000$ iterations each and plot the regret along with error bars. Comment on the variance of the algorithm. What happens to the variance when you play around with the shape parameters?. (Bonus: Can you come up with an algorithm which has a better variance than EXP3?)

Q3: ϵ-Greedy

An *epsilon*-greedy strategy for the stochastic multi-armed bandits set up exploits the current best arm with probability $(1 - \epsilon)$ and explores with a small probability ϵ. Consider a problem instance with 10 arms where the reward for the i-th ($i = 1, \ldots, 10$) arm is Beta distributed with parameters $\alpha_i = 5, \beta_i = 5 * i$. Implement the *epsilon*-greedy algorithm (by trying out various ϵ values)and compare it with the performance of the EXP-3 algorithm and an algorithm that chooses an arm in a greedy fashion at every round (i.e., $\epsilon = 0$) after playing each arm once in a round robin fashion. Plot the regret bounds and comment on your observations. (Bonus: Can you formally show a regret guarantee for the *epsilon*-greedy algorithm?)