Homework 2 – RL

The purpose of this exercise is to test and apply what you have learned to bandit armed-multi issues. A bank has considered a facility plan for some of its customers. In this plan, which includes students, civil servants and freelancers, the bank pays one of the three amounts of 5, 20 and 100 million tomans as a loan to the customer, and the customer is obliged to repay the principal amount plus Pay the service fee to the bank. If the customer is from any of the three mentioned categories, if he receives any of the amounts available for the facility, with a certain probability he will succeed in repaying the amount of that facility on the due date. These probabilities are given in the reward class in the codes provided to you (for example, students are likely to fail to repay the loan of 100 million tomans, but self-employed people mostly return this loan). The bank's policies should be It should be arranged to offer facilities that maximize the bank's profit according to the ability of different customers to repay the loan.
If he returns the provided services to the bank, the bank can use the fee received. Otherwise, the bank has suffered a loss equal to the difference between the amount paid and the amount returned. It is done by the customer. The fee for facilities of 5, 2 and 100 million tomans is 100 thousand, 750 thousand and 5 million tomans, respectively.

Questions

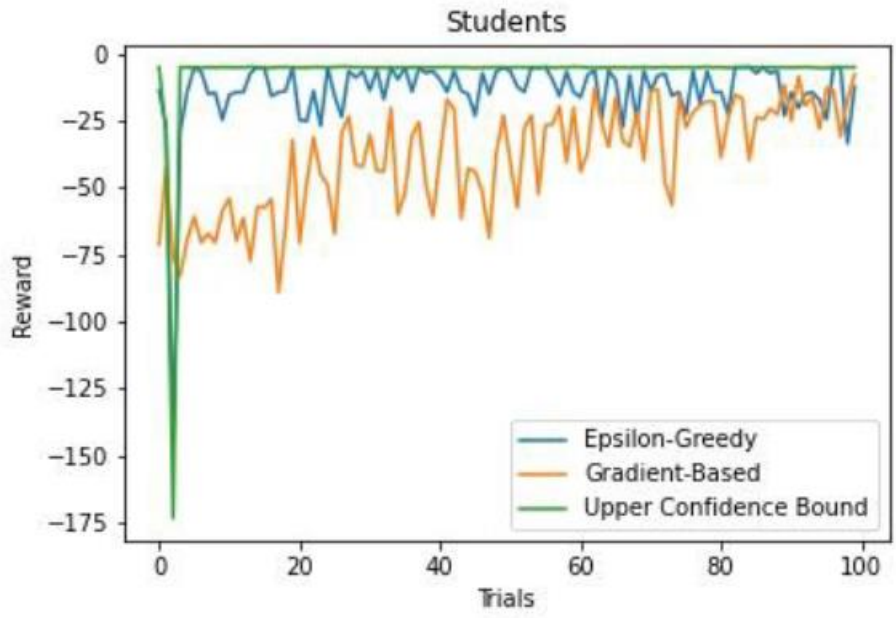
1. Present a model based on the Multi-Armed Bandit problem to maximize the bank's profit. According to the questions of the first part, it is necessary to explain the set of openings, rewards and how to answer the problem according to the different input states.

2. Implement the environment related to the model you presented in the previous section. For this, you can use the codes provided to you.

   Note that the operator is noted by the environment of the note value. Also, the set of arms is given to the agent through the environment, but the decision to choose between them is the responsibility of the agent.

3. Implement a Multi-Armed Bandit learning agent for each of the Epsilon-Greedy, Gradient-Based and Upper Confidence Bound algorithms. To avoid using duplicate code in the program, you can use the concepts of inheritance in object orientation. Make the implementation in such a way that the values of alpha, beta and gamma to calculate the utility function (Utility Function, suppose $u=\beta r^{\gamma}+\alpha$) as well as the hyperparameters required to execute the algorithm (such as epsilon) are given to the class as input to the constructor function.

4. Assuming that the utility function is equal to receiving from the arm, one sample of each of the three factors (example) determine the values of alpha, beta, and gamma so that it is assumed. Epsilon and equal to 0.2, point, fix the learning rate (assume equal to 0.001 and c equal to 2. Plot the received data and regret per trial for 20 runs of each algorithm with 100 trials. The final map of each table group should have three curves, each corresponding to an algorithm. For example, the resulting map from running the fish epsilon-greedy algorithm with epsilon equal to 0.1, grade-based gradient with a score of 0.005, and UCB with c equal to 4 in the corresponding student environment could look something like the image below.



5. It intends to find the most profitable facilities for each group of customers by providing different services to 60 people of the first offer bank. Assuming that there are equal numbers of each customer group in these 60 people, it is necessary to adjust the learning rate of the Gradient-Based algorithm in such a way that it can find the optimal facilities for each of the customer groups in the appropriate number of tests. For this, by examining 4 different values of the learning rate, draw a graph related to the average amount of reward received and the average amount of regret (Regret) by customer groups, as in the previous section (average in 20 executions). D. Choosing different values of the learning rate It is up to you, but you should finally determine the learning rate that you think is optimal (of course, the graph corresponding to this value should be one of the 4 curves in the final graphs).