CMU 10-715: Homework 10

Online Learning and Multi-armed Bandits

DUE: Dec. 6, 2020, 11:59 PM.

Instructions:

- **Collaboration policy:** Collaboration on solving the homework is allowed, after you have thought about the problems on your own. It is also OK to get clarification (but not solutions) from books, again after you have thought about the problems on your own. Please don't search for answers on the web, previous years' homeworks, etc. (please ask the TAs if you are not sure if you can use a particular reference). There are two requirements: first, cite your collaborators fully and completely (e.g., "Alice explained to me what is asked in Question 4.3"). Second, write your solution *independently*: close the book and all of your notes, and send collaborators out of the room, so that the solution comes from you only.
- **Submitting your work:** Assignments should be submitted as PDFs using Gradescope unless explicitly stated otherwise. Each derivation/proof should be completed on a separate page. Submissions can be handwritten, but should be labeled and clearly legible. Else, submission can be written in LaTeX.
- **Late days:** For each homework you get three late days to be used only when anything urgent comes up. No points will be deducted for using these late days. We will consider an honor system where we will rely on you to use the late days appropriately.

1 Majority Voting with Perfect Experts [25 pts]

We want to predict if the stock market will go up or down. On day t, we observe the actual outcome $y_t \in \{-1, 1\}$. Assume we have N experts who would vote positive $(+1)$ or negative (-1) on each day t. Consider the following online learning algorithm for this binary prediction problem:

Algorithm 1: Majority Voting Algorithm

Input: Pool of N experts: $S = \{1, \cdots, N\}$, number of days: T

for $t \leftarrow 1$ **to** T **do**

Let S_p^T and S_n^T be the non-overlapping subsets of S that voted positive and negative respectively on day t.

Make a prediction based on majority voting: $\hat{y}_t = -1 + 2 * \mathbb{1}\{|S_p^T| > |S_n^T|\}$

Observe the actual y_t

$S \leftarrow S_p^T$ if $y_t = +1$. Otherwise $S \leftarrow S_n^T$

end

On day t, the algorithm removes the experts that made a wrong prediction. Prove that if there exists at least a perfect expert who will always make the correct prediction, the algorithm will make at most $log_2 N$ mistakes. Assume that T is large enough. The number of mistakes is the number of days s.t. $\hat{y}_t \neq y_t$.

Solution: Whenever there is a mistake, as least half of S are removed. Also $|S| \geq 1$ due to the existence of the perfect expert. Hence we have $\frac{N}{2^M} \geq 1 \Rightarrow M \leq log_2 N$.

2 Decomposition of Regret [25 pts]

Consider a stochastic multi-armed bandit with K arms. Let μ_k be the mean of arm k for $k \in [K]$. Let $\mu^* = \max_{k \in [K]} \mu_k$. Recall that the (expected) regret is defined as:

$$R(T) = T\mu^* - \mathbb{E}[\sum_{t=1}^{T} R_t]$$

where R_t is the reward received at iteration t. Prove that regret can be rewritten in the following form:

$$R(T) = \sum_{k=1}^{K} \Delta_k \mathbb{E}[N_k(T)]$$

where $N_k(t)$ is the number of selections of arm k in the first t iterations, $\Delta_k = \mu^* - \mu_k$ is the sub-optimality gap of arm k.

Solution: Let $A_t \in [K]$ denote the selected arm at iteration t.

$$R(T) = T\mu^* - \mathbb{E}[\mu_{A_t}] = \mathbb{E}[\mu^* - \mu_{A_t}] = \sum_{k=1}^{K} \underbrace{(\mu^* - \mu_k)}_{\Delta_k} \mathbb{E}[\underbrace{\sum_{t=1}^{T} \mathbb{1}\{A_t = k\}}_{N_k(T)}]$$

3 Simulation of Bernoulli Multi-armed Bandits [50 pts]

In this question you will be running simulations on Bernoulli Multi-armed Bandits with different algorithms. Using the same notations as in Q2, assume the bandit has K arms whose indices $k \in [K]$ and the algorithm will run for T iterations. Consider the following three algorithms:

Algorithm 2: Follow the Leader (FTL)

$A = \{1, 2, \cdots, K\}$
for $t \leftarrow 1$ **to** K **do**
  Randomly select $a \in A$, receive the reward R_t.
  $\hat{\mu}_a \leftarrow R_t$
  $A \leftarrow A \setminus a$
end
for $t \leftarrow K + 1$ **to** T **do**
  $a = argmax_{k \in [K]} \hat{\mu}_k$
  Pull arm a and receive the reward R_t
  Update $\hat{\mu}_a$
end

Algorithm 3: Uniform Exploration

for $t \leftarrow 1$ **to** T **do**
  Uniformly select an arm $a \in [K]$
  Pull the arm and receive the reward R_t
end

Algorithm 4: Upper Confidence Bound (UCB)

$A = \{1, 2, \cdots, K\}$
for $t \leftarrow 1$ **to** K **do**
  Randomly select $a \in A$, receive the reward R_t.
  $\hat{\mu}_a \leftarrow R_t$
  $A \leftarrow A \setminus a$
end
for $t \leftarrow K + 1$ **to** T **do**
  $a = argmax_{k \in [K]} \hat{\mu}_k + \sqrt{\frac{2logt}{N_k(t-1)}}$
  Pull arm a and receive the reward R_t
  Update $\hat{\mu}_a$
end

To update $\hat{\mu}_a$, one simply computes the sample mean $\hat{\mu}_a = \frac{\sum R}{N_a(t)}$ where the numerator is the sum of the reward over first t iterations where arm a was selected and the denominator has the same meaning as in Q2. The three

algorithms (FTL, Uniform, UCB) represent pure exploitation, pure exploration, and exploration-exploitation trade-off respectively. Code the three algorithms with any programming language you prefer.

a (30 pts) Consider a Bernoulli bandit with **two** arms where $\mu_1 = 0.4, \mu_2 = 0.6$. Run each algorithm on this bandit with $T = 2000$ steps and 500 repetitions. On the same figure, plot the regret (averaged over 500 repetitions) vs iterations of the three algorithms. Also incldue the 5th and 95th percentile of the regret (out of the 500 repetitions) vs iterations. Briefly explain the difference in performance among the three algorithms.

b (20 pts) Now consider a Bernoulli bandit with **nineteen** arms where $\mu_k = 0.05k$. That is, $\mu_1 = 0.05, \mu_2 = 0.1, \cdots, \mu_{19} = 0.95$. Make the same plot as in (a). Compare the results with (a) and briefly discuss the difference.

Solution: Note that there are two different notions of regret: assume at one repetition, for $t = 1, 2$, we draw from arm 1 and 2 and receive reward 0 and 1 respectively. The **expected** regret uses the true means of the arm and will be $0.6 * 2 - (0.4 + 0.6)$. The **actual** regret uses the observed regret, and will be $0.6 * 2 - (0 + 1)$. Since we didn't specify which regret to use in the homework, either will be fine for this time. Fig 1 and 2 are the plots with the expected regret. Fig 3 and 4 are the plots with the actual regret. The conclusions you make should not really be affected by which type of regret you computed. In (a), uniform exploration achieves a linear regret since it continue select an arm randomly. FTL may easily get stuck on a worse arm ($\mu = 0.4$) if the first draw from arm 1 and 2 are $1, 0$ respectively, hence its percentile band is wide. UCB resolves this issue by allowing the algorithm to sample the other arm to avoid getting stuck. When the gap between the means are much smaller, FTL won't easily get stuck, whereas UCB may select a sub-optimal arm since the means are so close to each other.

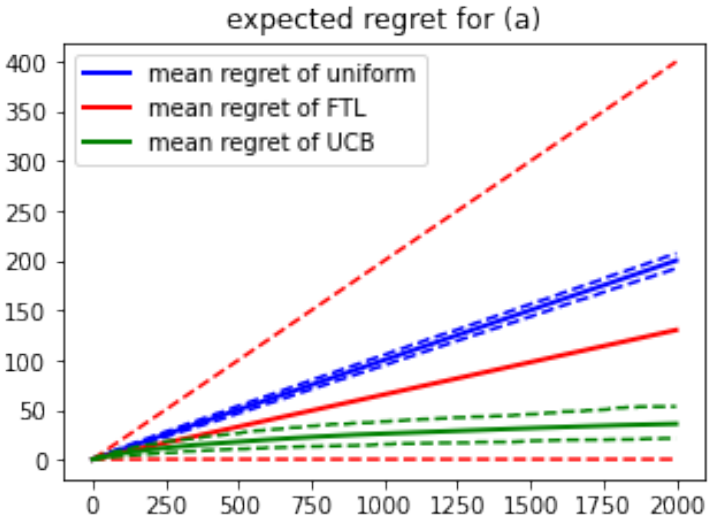

Figure 1: Expected regret for (a)

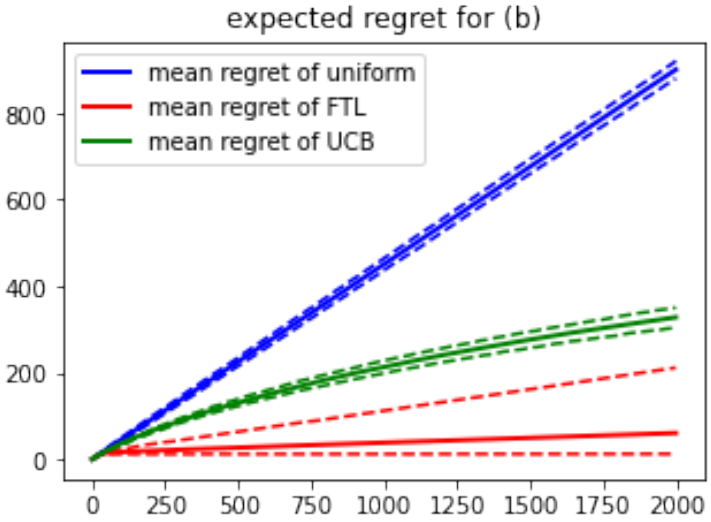

Figure 2: (Expected regret for (b)

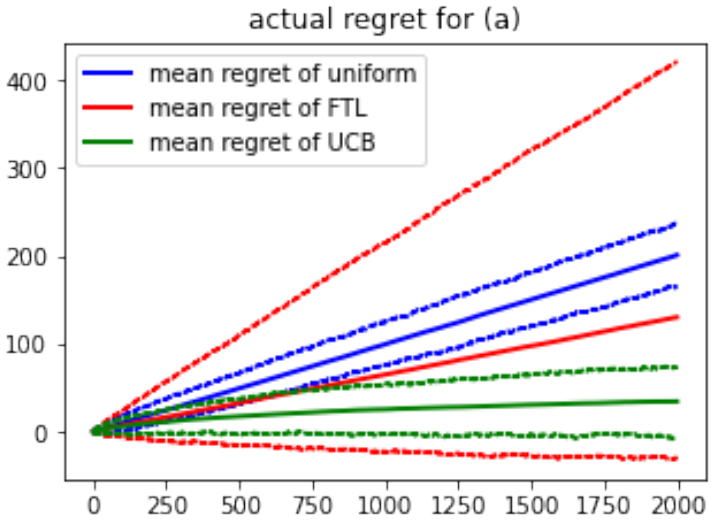

Figure 3: Actual regret for (a)

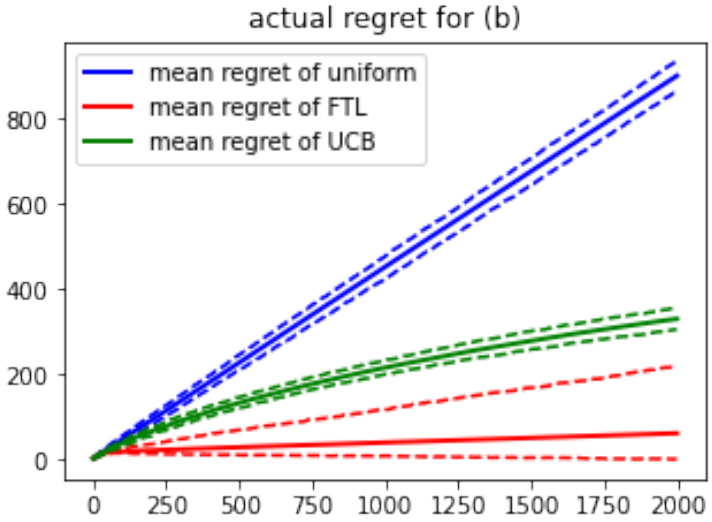

Figure 4: (Actual regret for (b)