TP on Adversarial Bandits

In the lecture, we saw the adversarial bandit framework as a game between a player and nature. In fact, there is a strong connexion between regret minimization and game theory. In this practical session, we will apply the EXP3 algorithm to a sequential two-player zero sum game.

We consider the sequential version of a two-player zero-sum games between a player and an adversary.

Let $L \in [-1,1]^{M\times N}$ be a loss matrix.
At each round $t = 1, \dots, T$
- The player choose a distribution $p_t \in \Delta_M := \{p \in [0,1]^M, \sum_{i=1}^M p_i = 1\}$
- The adversary chooses a distribution $q_t \in \Delta_N$
- The actions of both players are sampled $i_t \sim p_t$ and $j_t \sim q_t$
- The player incurs the loss $L(i_t, j_t)$ and the adversary the loss $-L(i_t, j_t)$.

Setting 1: Setting of a sequential two-player zero sum game

1. Define M, N and a loss matrix $L \in [-1,1]^{M\times N}$ that corresponds to the game "Rock paper scissors"[1].

Full information feedback

In this part, we assume that both players know the matrix L in advance and can compute $L(i,j)$ for any (i,j).

2. Implementation of `EWA`.
   (a) In order to implement the exponential weight algorithm, you need a way to sample from the exponential weight distribution. Implement the function `rand_exp` that takes as input a probability vector $p \in \Delta_M$ and uses a single call to `rand()` to return $X \in [M]$ with $P(X = i) = p_i$.
   (b) Define a function `EWA_update` that takes as input a vector $p_t \in \Delta_M$ and a loss vector $\ell_t \in [-1,1]^M$ and return the updated vector $p_{t+1} \in \Delta_M$ defined for all $i \in [M]$ by
$$p_{t+1}(i) = \frac{p_t(i)\exp(-\eta \ell_t(i))}{\sum_{j=1}^M p_t(j)\exp(-\eta \ell_t(j))} \,.$$
3. Simulation against a fixed adverary. Consider the game "Rock paper scissors" and assume that the adversary chooses $q_t = (1/2, 1/4, 1/4)$ and samples $j_t \sim q_t$ for all rounds $t \geq 1$.
   (a) What is the loss $\ell_t(i)$ incurred by the player if he chooses action i at time t? Simulate an instance of the game for $t = 1, \dots, T = 100$ for $\eta = 1$.

[1] This is a common game where two players choose one of 3 options: (Rock, Paper, Scissors). The winner is decided according to the following: Rock crushes scissors, Paper covers Rock, Scissors cuts paper

(b) Plot the evolution of the weight vectors $p_1, p_2, \dots, p_T$. What seems to be the best strategy against this adversary?

(c) Plot the average loss $\bar{\ell}_t = \frac{1}{t}\sum_{s=1}^t \ell(i_s, j_s)$ as a function of t.

(d) Plot the cumulative regret.

(e) To see if the algorithm is stable, repeat the simulation $n = 10$ times and plot the average loss $(\bar{\ell}_t)_{t\geq 1}$ obtained in average, in maximum and in minimum over the n simulations.

(f) Repeat one simulation for different values of learning rates $\eta \in \{0.01, 0.05, 0.1, 0.5, 1\}$ and plot the final regret as a function of η. What are the best η in practice and in theory.

4. Simulation against an adaptive adversary. Repeat the simulation of question 3) when the adversary is also playing EWA with learning parameters $\eta = 0.05$.

   (a) Plot $\frac{1}{t}\sum_{s=1}^t \ell(i_s, j_s)$ as a function of t.

   It is possible to show that if both players play according to a regret minimizing strategy the cumulative loss of the player converges to the value of the game

$$V = \min_{p\in\Delta_M} \max_{q\in\Delta_q} p^\top L q\,.$$

   (b) Define $\bar{p}_t = \frac{1}{t}\sum_{s=1}^t p_s$. Plot in log log scale $\|\bar{p}_t - (1/3, 1/3, 1/3)\|_2$ as a function of t.

   It is possible to show that $(\bar{p}_t, \bar{q}_t)_{t\geq 1}$ converges almost surely to a Nash equilibrium of the game. This means that if $p \times q$ is a Nash equilibrium, none of the players should change is strategy if the other player does not change hers.

Bandit feedback

Now, we assume that the players do not know the game in advance but only observe the performance $L(i_t, j_t)$ (that we assume here to be in $[0, 1]$) of the actions played at time t. They need to learn the game and adapt to the adversary as one goes along.

5. Implementation of `EXP3`. Since both players are symmetric, we focus on the first player.

   (a) Implement the function `estimated_loss` that takes as input the action $i_t \in [M]$ played at round $t \geq 1$ and the loss $L(i_t, j_t)$ suffered by the player and return the vector of estimated loss $\widehat{\ell}_t \in \mathbb{R}_+^M$ used by `EXP3`.

   (b) Implement the function `EXP3_update` that takes as input a vector $p_t \in \Delta_M$, the action $i_t \in [M]$ played by the player and the loss $L(i_t, j_t)$ and return the updated weight vector $p_{t+1} \in \Delta_M$.

6. Repeat Questions 3.a) to 3.f) with `EXP3` instead of `EWA`.

7. Repeat Question 4.a) and 4.b) with `EXP3` instead of `EWA`.

Optional extentions

8. Repeat Question 4.a) when the adversary is playing a `UCB` algorithm. Who wins between `UCB` and `EXP3`?

9. In this lecture, we saw that EXP3 has a sublinear expected regret. Yet, as shown by question 6.e), it is extremely unstable with a large variance. Implement `EXP3.IX` (see Chapter 12 of [1]) a modification of `EXP3` that controls the regret in expectation and simultaneously keeps it stable. Repeat question 3.e) with `EXP3.IX`

10. Try different games (not necessarily zero-sum games). In particular, how these algorithms behave for the prisoner's dilemna (see wikipedia)? The prisoner's dilemna is a two-player games that shows why two completely rational individuals might not cooperate, even if it appears that it is in their best interests to do so. The losses matrices are:

$$L^{(player)} = \begin{pmatrix} 1 & 3 \\ 0 & 2 \end{pmatrix} \quad \text{and} \quad L^{(adversary)} = \begin{pmatrix} 1 & 0 \\ 3 & 2 \end{pmatrix} .$$

References

[1] Tor Lattimore and Csaba Szepesvári. Bandit algorithms. `https://tor-lattimore.com/downloads/book/book.pdf`, 2019.