Contextual Bandits

Henry Vu

Feb 29, 2024

Recap: Exp3 Algorithm

Algorithm 3 EXP3 Algorithm

1: **Input:** horizon n, number of arms K, learning rate η;

2: Set $\hat{S}_{0,i} = 0$ for all $1 \leq i \leq K$;

3: **for** $t = 1, 2, \ldots, n$ **do**

4: $\quad P_{t,i} = \frac{\exp(\eta \hat{S}_{t-1,i})}{\sum_{j \in [k]} \exp(\eta \hat{S}_{t-1,j})}$;

5: $\quad$ Sample $A_t \sim P_t$, receive X_t;

6: $\quad$ Update $\hat{S}_{t,i} = \hat{S}_{t-1,i} + 1 - \frac{\mathbb{I}(A_t = i)(1 - X_t)}{P_{t,i}} = \hat{S}_{t-1,i} + \hat{X}_{t,i}$;

7: **end for**

Exp3 Algorithm

- **Main idea**: Construct an estimator for the loss functions seen in the Hedge algorithm (in this setting, reward). 2 candidates:

- Candidate 1: $\hat{X}_{t,i} := \frac{\mathbf{1}\{A_t = i\} \cdot x_{t,i}}{P_{t,i}} = \frac{\mathbf{1}\{A_t = i\} \cdot x_{t,A_t}}{P_{t,A_t}} = \frac{\mathbf{1}\{A_t = i\} \cdot X_t}{P_{t,A_t}}$

  Unbiased estimator, since

$$\mathrm{E}[\hat{X}_{t,i}|\mathcal{H}_{t-1}] = \mathrm{E}\left[\frac{\mathbf{1}\{A_t = i\} \cdot X_t}{P_{t,i}}\right] = \frac{P_{t,i} \cdot x_{t,i}}{P_{t,i}} = x_{t,i}$$

- $1\{A_t=i\}X_t = 1\{A_t=i\}x_{t,i}$ and $x_{t,i}$ is a function of history $f(H_{t-1})$ which is non-random if given H_{t-1}. $P_{t,i}$ is also a function of history $g(H_{t-1})$ and is deterministic given history.

Recap: Exp3 Algorithm

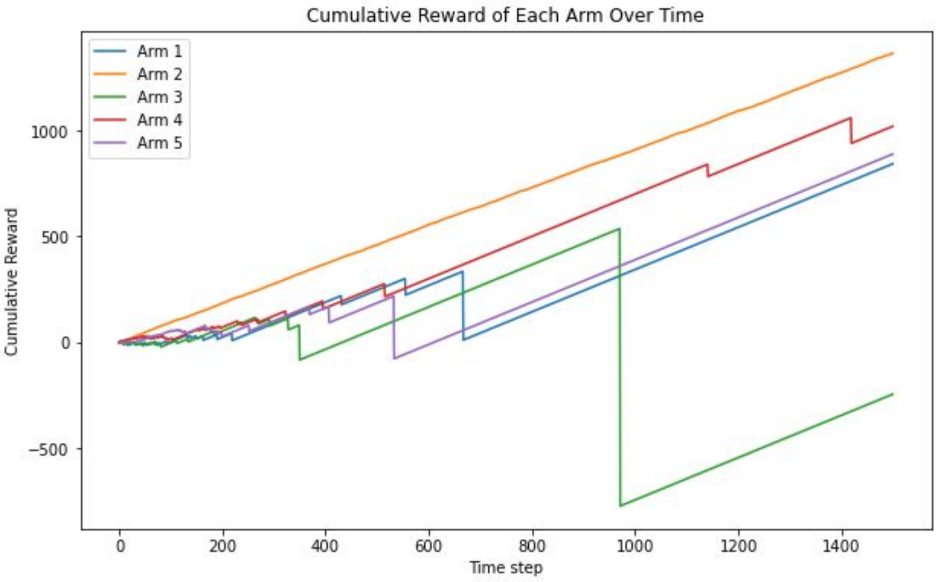
- Another alternative is to use

$$\hat{X}_{t,i} := 1 - \frac{\mathbf{1}\{A_t = i\}}{P_{t,i}} \cdot (1 - X_t) = 1 - \frac{\mathbf{1}\{A_t = i\}}{P_{t,i}} \cdot (1 - x_{t,i})$$

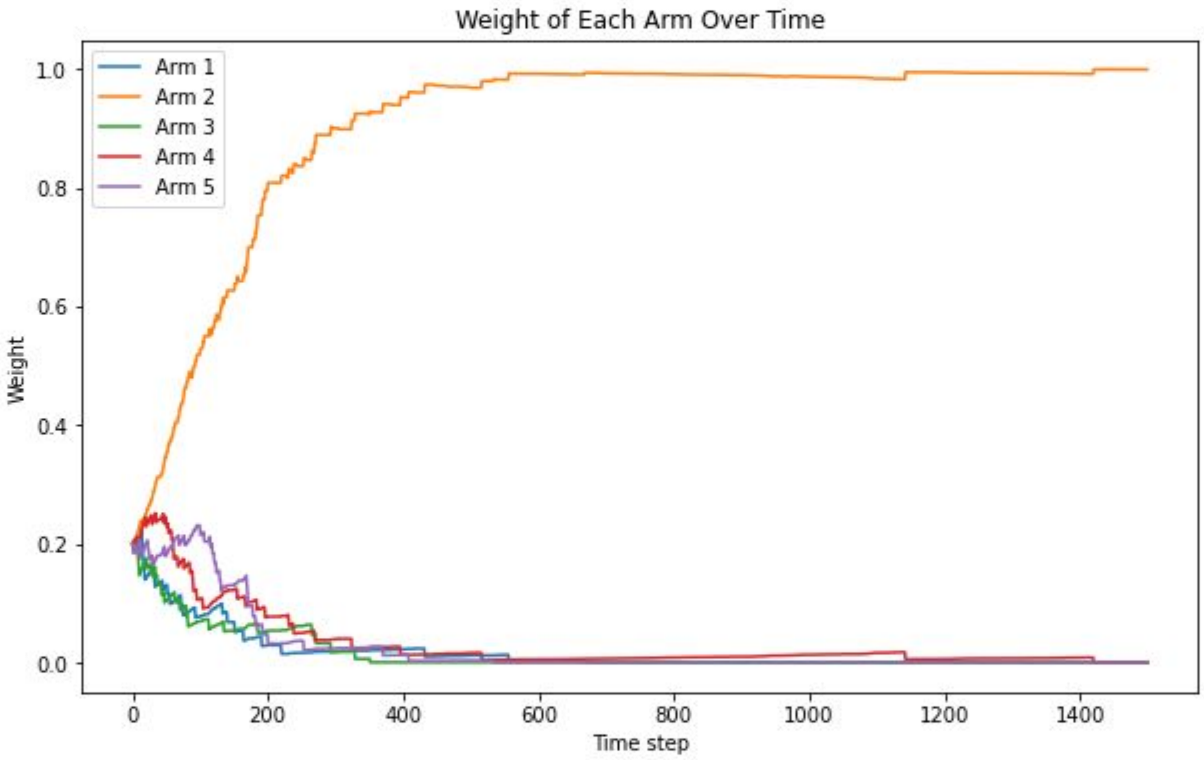
- This unbiased estimator is an interpretation of “loss” and takes on values in $(-\infty, 1]$.

Implementation Results

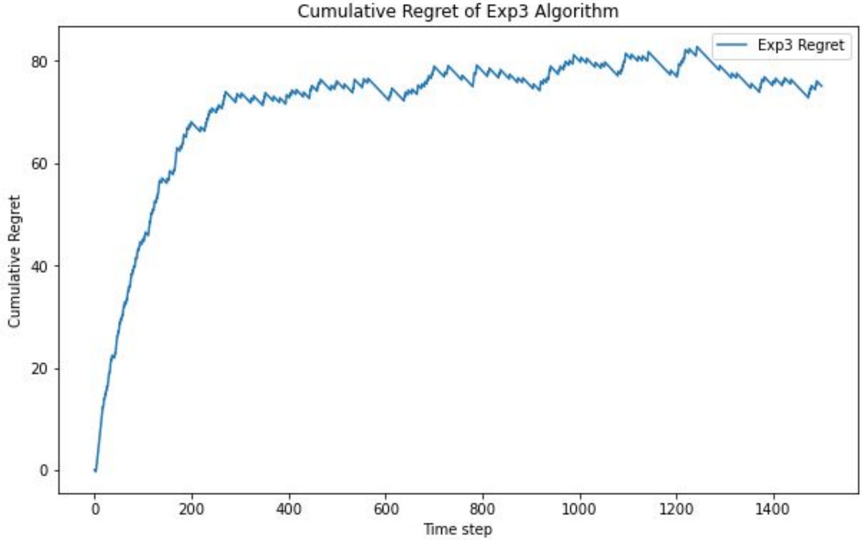
- Time horizon n = 1500, number of arms K = 5
- Each arm has a bernoulli reward distribution with P = [0.1, 0.9, 0.2, 0.3, 0.15]

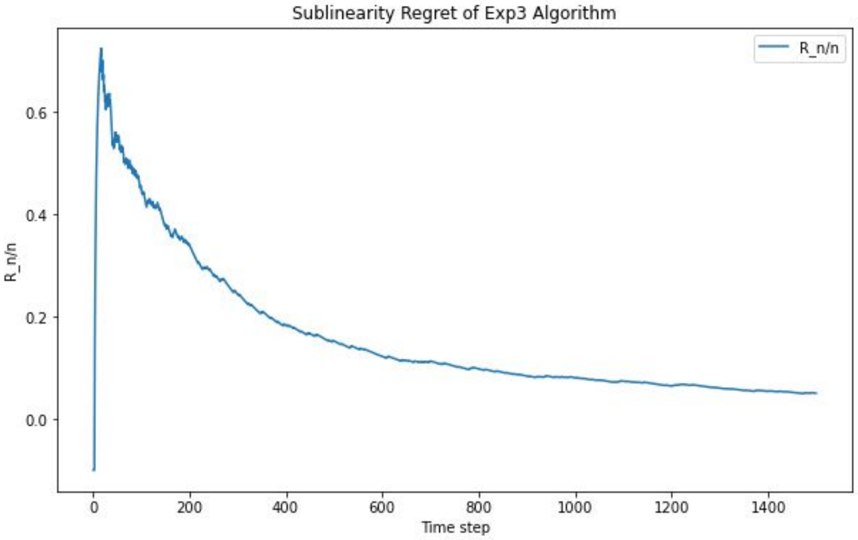


Probability of Arm Over Time



Regret of Exp3





Theorem 1 (Lattimore. Theorem 11.2). For rewards $x_{t,i} \in [0,1]$, and the learning rate tuned to $\eta = \sqrt{2\log(K)/(nK)}$, we have for any arm i

$$R_{n,i} \leq \sqrt{2nK\log(K)}$$

Today's Agenda: Contextual Bandits

- In many applications, the context provides additional information for selecting an action
  - E.g., personalized advertising, user interfaces
  - Context: location, age, gender, preference, etc.

- The most basic example of contextual bandits is obtained when rounds $t = 1, 2, \ldots$ are marked by contexts $c_1, c_2, \ldots$ from a given finite context set C. The forecaster must learn the best mapping $\Phi: C \to [K]$ of contexts to arms.

$\Rightarrow$ Context as a notion of *states*

Contextual Bandits: Reward

- The best total reward after n rounds if we can adjust actions to contexts is:

$$
\begin{aligned}
S_n &= \sum_{c \in C} \max_{k \in [K]} \sum_{t: c_t = c} x_{t,k} \\
&= \max_{\phi: C \to [K]} \sum_{t=1}^{n} x_{t,\phi(c_t)}.
\end{aligned}
$$

- The regret after n rounds is the sum of the regrets suffered by the bandits assigned to the individual contexts c_t

$$R_n = S_n - \sum_t X_t = \sum_{c \in C} \mathbb{E}\left[\max_{k \in |K|} \sum_{t: c_t = c} (x_{t,k} - X_t)\right]$$

Solving Contextual Bandits

- Let $T^c(n)$ be the number of occurrences of context c in n rounds.

- Let $R^c(s)$ be the regret of the instance of Exp3 associated with c at the end of the round when this instance is used s times. We can write regret as

$$R_n = \sum_{c \in C} \mathbb{E}\left[R^c\left(T^c(n)\right)\right]$$

- $T^c(n)$ may vary from context to context, we shouldn't use a version of Exp3 tuned for a fixed number of rounds

Solving Contextual Bandits

- Idea: Choose parameter η of Exp3 that depends on round index. When an Exp3 instance is used the s'th time, set

  $\eta_s = \sqrt{\frac{\log(K)}{sK}}$ to obtain regret $R^c(s) \leq 2\sqrt{sK\log(K)}$ for any s

- One can show that

$$R_n = \sum_{c \in C} \mathbb{E}\left[R^c\left(T^c(n)\right)\right]$$
$$\leq \sqrt{2|C|nK \ln K}$$

Contextual Bandits with Exp3

- By Jensen's inequality, since $f(x)=x^2$ is convex when $x \geq 0$,

$$
\begin{aligned}
R_n &= \sum_{c \in C} \mathbb{E}\left[R^c\left(T^c(n)\right)\right] \\
&\leq \sum_{c \in C} \sqrt{2T^c(n) K \ln K} \\
&= \sqrt{2K \ln K} \sum_{c \in C} \sqrt{T^c(n)} \\
&\leq \sqrt{2K \ln K} \sqrt{|C| \sum_{c \in C} (\sqrt{T^c(n)})^2} \\
&= \sqrt{2|C| n K \ln K} \qquad \text{since } \sum_c T^c(n) = n
\end{aligned}
$$

Large Context Space?

- When the context set C is large, using one instance of Exp3 for each context is a poor choice.
- Idea: Group contexts with similar internal structure and assign a bandit to each group

Before

$$S_n = \sum_{c \in C} \max_{k \in [K]} \sum_{t : c_t = c} x_{t,k}$$
$$= \max_{\phi : C \to [K]} \sum_{t=1}^{n} x_{t,\phi(c_t)}.$$

After

$$S_n = \sum_{P \in \mathcal{P}} \max_{k \in |K|} \sum_{t : c_t \in P} x_{t,k}$$
$$= \max_{\phi \in \Phi(P)} \sum_{t=1}^{n} x_{t,\phi(c_t)}$$

Grouping

- Regret is in the same form

$$
\begin{aligned}
R_n &= \sum_{c \in C} \mathbb{E}\left[R^c\left(T^c(n)\right)\right] \\
&\leq \sum_{c \in C} \sqrt{2T^c(n)K \ln K} \\
&= \sqrt{2K \ln K} \sum_{c \in C} \sqrt{T^c(n)}
\end{aligned}
$$

but with $c \in C$ changed to $p \in P$ and the definition of $T^c(n)$ adjusted accordingly

Supervised Learning

- Another idea: run a supervised learning algorithm trained on some batch data to find a few predictors (classifiers) $\Phi_1, \dots, \Phi_M: C \rightarrow [K]$.

- In any case, we have a set of functions $\boldsymbol{\Phi}$ with the goal of competing with the best of them
  $\Rightarrow$ Think more generally about some subset $\boldsymbol{\Phi}$ of functions without necessarily considering the internal structure of contexts
  $\Rightarrow$ This viewpoint leads to *bandits with expert advice*

Contextual Bandits with Expert

- At the beginning of each round, M experts announce their predictions of which actions are the most promising.

- For the sake of generality, allow the experts to report not only a single prediction but a probability distribution over the actions.
  An interpretation is that the experts want to randomize their actions.

- We collect the advice of M experts for round t into an M x K matrix $E^{(t)}$
  $E^{(t)}_m$ is the probability distribution that expert m recommends for round t.

Next Week

Exp 4 Algorithm

Thank you!