MABs with Correlated Arms

EE6106 Course Project

Aziz Shameem Sameep Chattopadhyay

Guides: Prof. D Manjunath & Prof. Jayakrishnan Nair

IIT Bombay

April 2023

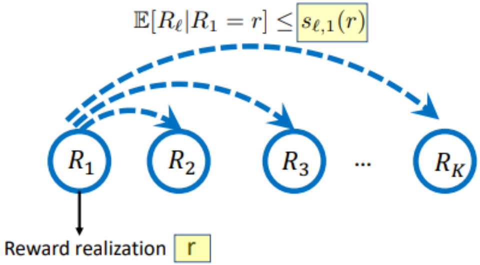
Flow

1. Introduction
2. Problem Formulation
3. Correlated MAB Setting
4. Algorithms Implemented
5. Experimental Results

Introduction

- The classical MAB algorithms implicitly assume that the rewards generated by the arms are uncorrelated to one another; pulling one arm provides no new information about the rest K-1 arms.
- Several Areas of practice - treatments/drugs/ad-versions, arms likely to be correlated, and the assumption does not hold true.
- Based on the correlated MAB setup in [1], we have utilized the correlation to improve the performance of 6 algorithms and have used them to build a movie recommender system tested on real-life dataset

Problem Formulation



The correlation between arms is captured by pseudo-rewards, which are indicative of an upper bound on the true mean of other arms.
We leverage these to reduce the number of arms dealt with at each time step.
Note : For rewards bounded between $[A, B]$, a pseudo-reward of B indicates an unconstrained version of the problem setting.

Problem Formulation: Global Recommender

- System has no access to contextual features of user eg-age, gender, income, etc. Hence cannot provide personalized recommendation
- Aims to provide global recommendations to a population with unknown demographics. We have built a global movie recommender
- Intuitively, a user reacting positively to movie A might also be more likely to react positively to the movie B belonging to same genre or having same lead actor
- Pseudo-rewards in this case would be an upper bound on the rating of movie B based on the user's rating of movie A

Correlated MAB Setting: Computing Pseudo Rewards

Methods-

- The pseudo-rewards can be learned through offline surveys in which users are allowed sample the rewards jointly
- In the presence of a training dataset, it can be computed using the empirical mean for joint distribution ($\hat{\mu}_{l,k}(r)$)
- In case of insufficient data, use the maximum possible reward for the corresponding arm

After t rounds, arm k is pulled $n_k(t)$ times. Using this $n_k(t)$ reward realizations, we can construct the empirical pseudo-reward ($\hat{\phi}_{l,k}(t)$) for each arm l w.r.t. arm k as follows-

$$\hat{\phi}_{l,k}(t) \triangleq \frac{\sum_{\tau=1}^{t} \mathcal{I}_{k_\tau = k}(r_{k_r}) s_{l,k}(r)}{n_k(t)}$$

Correlated MAB Setting: Correlated Algorithms

The procedure for utilizing the correlated MAB setting for any generalized bandit algorithm-

1. **Identify Significant Arms:** At each round t, define-
   S_t $=\{$ $k \in \mathcal{K}$: $n_k(t) \geq$ t/K$\}$.
   $k^m(t) =$ argmax$_{k \in S_t}$ $\hat{\mu}_k$ & $\hat{\mu}_{k^m}(t) =$ max$_{k \in S_t}$ $\hat{\mu}_k$
2. **Identify Competitive Arms:** Use $\hat{\mu}_{k^m}(t)$ to define non-competitive arms; an arm k is said to be Non-Competitive at round t, if-

   $$min_{l \in S_t}(\hat{\phi}_{l,k}(t)) \leq \hat{\mu}_{k^m}(t)$$

   $min_{l \in S_t}(\hat{\phi}_{l,k}(t))$ provides the tightest estimated upper bound on the mean rewards; if it is smaller than $\hat{\mu}_{k^m}(t)$, then the arm seems unlikely to be optimal.
3. **Play Algorithm on Competitive Arms:** Now play the original MAB algorithm on the set of optimal arms (k $\notin$ S_t)

Algorithms : Epsilon Greedy

- With probability ϵ : Sample Arms Uniformly
- With probability $1-\epsilon$: Choose Arm $A_{t+1} = \text{argmax}_k(\hat{\mu}_k(t))$

Vanilla Epsilon Greedy

$$\hat{\mu}_k(t+1) = \hat{\mu}_k(t) + \frac{R_t - \hat{\mu}_k(t)}{n_k(t)} * \mathcal{I}_{A_t=k}$$

Epsilon Greedy : Constant Learning Rate

$$\hat{\mu}_k(t+1) = \hat{\mu}_k(t) + \alpha(R_t - \hat{\mu}_k(t))\mathcal{I}_{A_t=k}$$

Epsilon Greedy : Optimistic Initial Values

$\hat{\mu}_k(0) =>$ largest possible value of reward

$$\hat{\mu}_k(t+1) = \hat{\mu}_k(t) + \frac{R_t - \hat{\mu}_k(t)}{n_k(t)+1} * \mathcal{I}_{A_t=k}$$

Algorithms : Gradient Bandit

Maintain *preferences* : $H_t(a)$

Preference has no interpretation in terms of reward. Only the relative preference of one action over another is important;

Arms chosen according to :

$$\mathrm{P}(A_t = a) = \frac{e^{H_t(a)}}{\sum_{b=1}^{k} e^{H_t(b)}}$$

Update rule :

$$H_{t+1}(A_t) = H_t(A_t) + \alpha(R_t - \bar{R}_t)(1 - \mathrm{P}(A_t = A_t))$$
$$H_{t+1}(a) = H_t(a) - \alpha(R_t - \bar{R}_t)(\mathrm{P}(a)) \text{ for } a \neq A_t$$

where $\alpha > 0$ is a step-size parameter, and $\bar{R}_t \in R$ is the average of all the rewards up through and including time t, which can be computed incrementally.

Algorithms : UCB

Start by choosing each arm once, in a round robin.
Then, choose arms satisfying :

$$A_t = \underset{a}{\operatorname{argmin}} \left[Q_t(a) + \sqrt{\frac{lnt}{N_t(a)}} \right]$$

where $N_t(a)$ denotes the number of times that action a has been selected prior to time t, and the number $c > 0$ controls the degree of exploration. $Q_t(a)$ can correspond to the empirical means of the observations upto time t.

Algorithms : Thompson Sampling

Under Thompson sampling, the arm $k_{t+1} = \text{argmax}_{k \in \mathcal{K}} S_{k,t}$ is selected at time step $t+1$. Here, $S_{k,t}$ is the sample obtained from the posterior distribution of μ_k, That is,

$$k_{t+1} = \underset{k \in \mathcal{K}}{\text{argmax}} \, S_{k,t}$$
$$S_{k,t} \sim \mathcal{N}\left(\hat{\mu_k}(t), \frac{\beta B}{n_k(t)+1}\right)$$

here *beta* is a hyperparameter for the Thompson Sampling algorithm, and B denotes the maximum possible reward that can be obtained (an upper bound on the reward distribution)

Experiments and Results

- We have performed tasks of genre recommendation & movie recommendation for a subset of the real-life Movielens dataset.
- The dataset contains the movie ratings by users on a scale of 1-5, the subset used for experiment has 50 movies spread across 18 genres
- For the task, the movies/genre are treated as arms, with the user ratings being their corresponding rewards.

Experiments and Results : Genre Recommendation

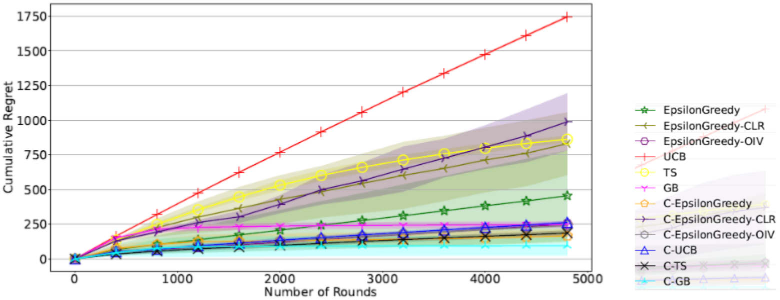


Figure: Plot showing the evolution of regret with the number of pulls

Experiments and Results : Genre Recommendation

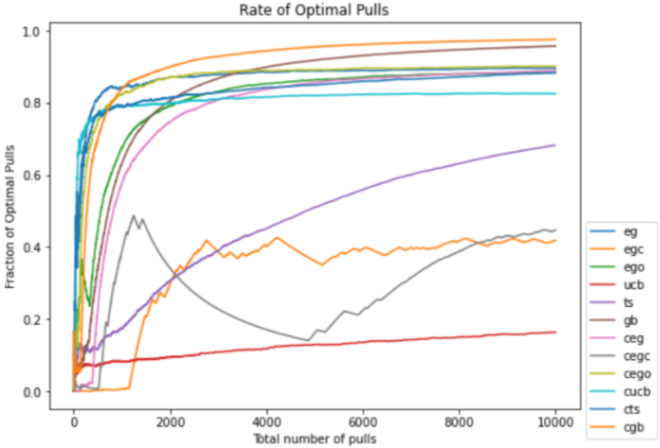


Figure: The fraction of optimal pulls against the total number of arm pulls

Experiments and Results : Movie Recommendation

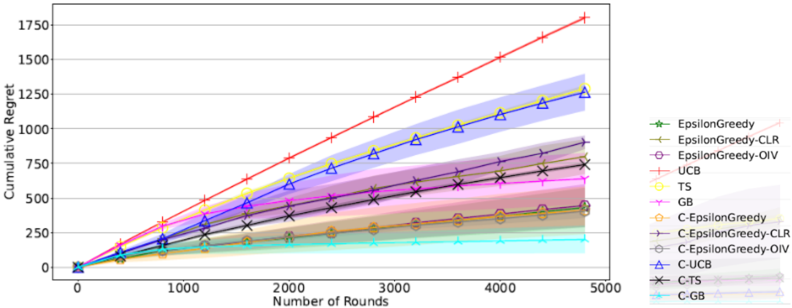


Figure: Plot showing the evolution of regret with the number of pulls

Experiments and Results : Movie Recommendation

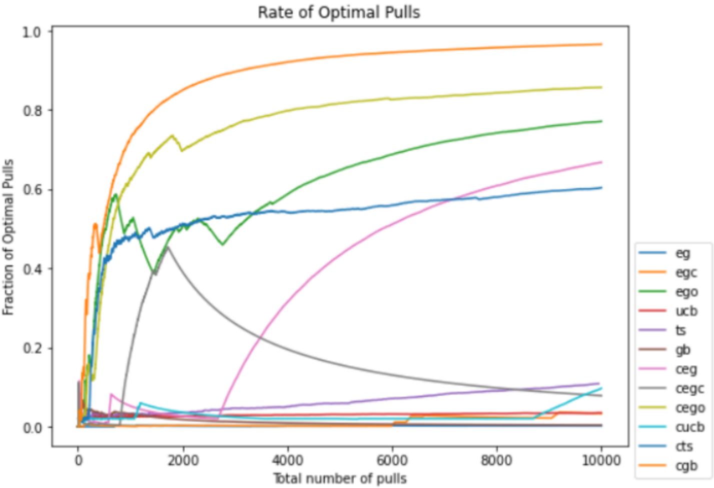
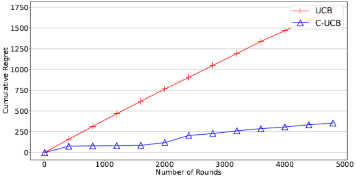
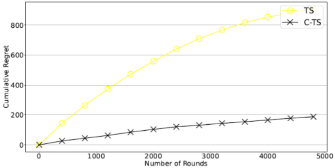
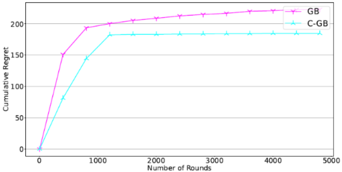


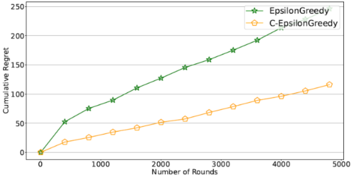
Figure: The fraction of optimal pulls against the total number of arm pulls

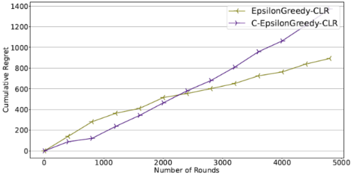
Comparison of Algos with their C-versions

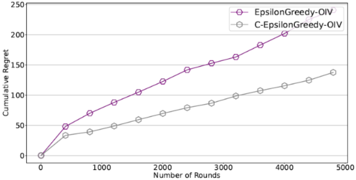












References

- S. Gupta., S. Chaudhari, G. Joshi, O. Yagan : Multi-Armed Bandits with Correlated Arms
- Richard Sutton, Andrew Barto : Reinforcement Learning, An Introduction
- F. M. Harper and J. A. Konstan, “The movielens datasets: History and context,” ACM Transactions on Interactive Intelligent Systems (TiiS), vol. 5, 4, Article 19, 2015.
- https://towardsdatascience.com/thompson-sampling-fc28817eacb8