Research Reading Assignment

of the paper titled
"Correlated Bandits for Dynamic Pricing via the ARC algorithm"
by
Samuel N. Cohen and Tanut Treetanthiploet

Prepared by,
Subhrajyoty Roy
MB1911

Assignment for Computational Finance course
as a part of Master of Statistics curriculum

Session: 2020-21
Indian Statistical Institute, Kolkata

March 23, 2021

1 Introduction

Multi-armed bandit problem [3] depicts the situation of a gambler who enters into a casino with a finite amount of money and is posed with k slot machines, each of which has a different probability of winning. The gambler is faced with the problem of allocating the resources efficiently to maximize his (her) expected payoff in the long run. The exploration-exploitation dilemma convolutes the problem: if the gambler finds a profitable slot machine and pours his all money into that, he is potentially losing the chances of gaining more from some other slot machines which he has not explored enough. This problem, originated from a theoretical probability background, has enormous real-life applications.

1. A company has to understand which of the k types of ads (or banners) to display to attract its customers and gain long term popularity. Here, the constant exploitation of a single banner might make it mundane for the customers.

2. An exchange needs to allocate the underlying stock or assets among available k types of financial instruments to make available for customers to purchase, to maximize its revenue.

3. A person needs to allocate its resources (or money) to purchase k different instruments on possibly different underlying, to maximize his gain. This is a typical portfolio optimization problem, which is dynamic in nature.

2 Mathematical Formulation of the Problem

Multi-armed bandit problem starts with a space of actions $\mathcal{A} = \{1, 2, \dots K\}$. At time t, the gambler chooses action $A_t \in \mathcal{A}$, and if $A_t = k$, then he / she observes a random variable $Y_t^{(k)}$ and obtain a reward of $R^{(k)}(Y_t^{(k)})$, where the reward function may be different for different actions. There also exist a latent parameter Θ which specifies the distribution of all $\{Y_t^{(k)}\}_{k=1,t=1}^{K,\infty}$ such that, $Y_t^{(1)}, \dots Y_t^{(K)}$ are conditionally independent given Θ for every t. An additional randomization over the action space $\mathcal{A}$ can be used to allow mixed strategies, by considering a sequence $(U_t)_{t=1}^{\infty}$ taking values in $\Delta^k = \{(u_1, \dots u_K) : \sum_{i=1}^{K} u_i = 1 \text{ and } u_i \geq 0 \forall i\}$. Here, i-th component of U_t denotes the probability of choosing action i at round t.

There is a mutlitude of objective functions that one might choose to optimize with respect to the choice of U_t such as

1. Cumulative regret, $\mathcal{R}(\Theta, T, U_t) = \sum_{t=1}^{T} \left(\arg\max_k \mathbb{E}(R^k(Y_t^{(k)})) - \mathbb{E}_{U_t}(R^{(A_t)}(Y_t^{(A_t)})) \right)$.

2. Discounted reward, $V(\Theta, \beta, U_t) = \sum_{t=1}^{\infty} \beta^{(t-1)} \mathbb{E}_{U_t}(R^{(A_t)}(Y_t^{(A_t)}))$.

In the specific example of dynamic pricing considered by the authors [7], an exchange (or a store) has to determine which financial contracts to make available to maximize its revenue, let at the start of the day, the exchange needs to choose one of the contracts having features from the set $\{x_1, x_2, \dots x_K\}$. On day t, with a chosen contract with feature x_k, $N_t^{(k)}$ customers visit the website of the exchange and $p(x_k)$ is the demand probability that each customer buys the contract. Assuming a GLM model for this probability yields

$$Q_{i,t}^{(k)} \mid \Theta \sim_{IID} h(q) \exp\left(\phi(\Theta^\intercal x_k)q - G(\phi(\Theta^\intercal x_k))\right), \dots i = 1, 2, \dots N_t^{(k)}$$

which takes value 0 or 1 depending on whther i-th customer buys the contract or not. Assume, h, ϕ and G are some known functions. Finally, a reward $R^{(k)}(\Theta, N_t^{(k)}, \sum_{i=1}^{N_t^{(k)}} Q_{i,t}^{(k)})$ is collected by the exchange which depends on both the number of persons visiting exchange website and the number of contracts sold.

3 Solution to the Problem

In each of the step t, there are two subproblems. In the first one, the store must make an inference of Θ given the history upto time $(t-1)$, i.e. $\mathcal{F}_{t-1}$. In the second phase, based on this inference, the store must take an action A_t which minimizes either the cumulative regret or maximizes the discounted reward.

3.1 First Step: Inference about Θ

To solve the first step, assuming that the link function ϕ is differentiable, we define $\psi = (G' \circ \phi)^{-1}$ and note that by central limit theorem,

$$\sqrt{n}\left(\frac{1}{n}\sum_{i=1}^{n} Q_{i,t}^{(k)} - \Theta^\intercal x_k\right) \xrightarrow{d} \mathcal{N}\left(0, 1/[G''(\phi(\Theta^\intercal x_k))\phi'(\Theta^\intercal x_k)^2]\right), \; n \to \infty \tag{1}$$

Using eqn. (1), one can show that $\Theta \mid \mathcal{F}_{t-1} \sim \mathcal{N}(M_{t-1}, \Sigma_{t-1})$ where the posterior estimates of M_t, Σ_t are updated by observing $N_t^{(k)}, Q_{i,t}^{(k)}$ as follows

$$M_t = M_{t-1} + R_t^{(k)}(\Psi_t^{(k)} - M_{t-1}^\intercal x_k)\Sigma_{t-1}x_k \tag{2}$$

$$\Sigma_t = \Sigma_{t-1} - R_t^{(k)}\Sigma_{t-1}x_k x_k^\intercal \Sigma_{t-1} \tag{3}$$

where $\Psi_t^{(k)} = \frac{1}{n}\sum_{i=1}^{n} Q_{i,t}^{(k)}$ and $R_t^{(k)} = N_t^{(k)}V_t^{(k)}/(N_t^{(k)}V_t^{(k)} + x_k^\intercal \Sigma_{t-1}x_k + 1)$, $V_t^{(k)} = [G''(\phi(\Theta^\intercal x_k))\phi'(\Theta^\intercal x_k)^2]$. The eqn. (2)-(3) are simply Kalman filter type equations.

3.2 Second Step: Taking optimized action

There are numerous algorithms to maximize the reward in the long run, given the inference about the latent parameter has already been performed.

1. Greedy algorithm to always take action $k = \arg\max \mathbb{E}_{\Theta \mid \mathcal{F}_{t-1} \sim \mathcal{N}(M_{t-1}, \Sigma_{t-1})} R^{(k)}(\Theta, N_t^{(k)}, \sum_i Q_{i,t}^{(k)})$ may lead to suboptimal solution [5].

2. ϵ-greey algorithm [8] takes the greey action with probability $(1-\epsilon)$, and chooses any other action at random with probability ϵ. This ensures a balance between exploring results of new actions to ensure the inference about Θ is always updated.

3. Explore-then-commit algorithm [4] explores randomly for $[\epsilon T]$ rounds which is only used to update inference about Θ, and then it keeps taking the greey action only.

4. Thompson sampling [6] calculates the probability $\arg\max R^{(k)}(\Theta, N_t^{(k)}, \sum_{i=1}^{N_t^{(k)}} Q_{i,t}^{(k)}) = k$ for every choice of $k \in \mathcal{A}$. In other words, it calculates the probability under $\Theta \mid \mathcal{F}_{t-1}$ that action 1 is best, action 2 is best and so on. Then, it samples an action k according to that probability distribution. In other words, Thompson sampling reverses the maximization and expectation step in greedy algorithm.

There are some other algorithms as well. The authors [7] presents an improved **Asymptotic Randomized Control (ARC)** algorithm as another alternative to take this optimized action.

3.3 Asymptomatic Randomized Control Algorithm

Asymptomatic Randomized Control (ARC) algorithm [1] optimizes the choice of action a to maximize the discounted reward (instead of minimizing cumulative regret). Denoting the term $f = [\mathbb{E}(R^{(k)}(Y_t^{(k)}) \mid \mathcal{F}_{t-1})]_{k=1,2,\dots K}$ and $L^\lambda(a)$ as a special exploration term, the maximization of discounted reward leads to the solution of the fixed point equation as

$$a = f(a) + \frac{\beta}{(1-\beta)} L^\lambda(a), \quad \text{where } a \in \Delta^k = \{(u_1, \dots u_K) : \sum_{i=1}^{K} u_i = 1, u_i \geq 0\}$$

where $f(a)$ is the expected reward over one time step if the action is chosen according to the distribution $\nu^\lambda(a) = (\nu_1^\lambda(a), \dots \nu_K^\lambda(a))$ over the action space $\mathcal{A}$, where $\nu_i^\lambda(a) = \exp(a_i/\lambda)/\sum_j \exp(a_j/\lambda)$. The choice of $L^\lambda(a)$ is used to motivate exploration rather than only committing towards the optimal action, in order to update inference about Θ over time. The expression of $L^\lambda(a)$ is given by,

$$\begin{aligned} L_k^\lambda(a, m, \Sigma) := \langle \mathcal{B}^\lambda(a, m, \Sigma); \mathbb{E}_{m,\Sigma}\big(\Delta\Sigma^{(k)}\big)\rangle + \langle \mathcal{M}^\lambda(a, m, \Sigma); \mathbb{E}_{m,\Sigma}\big(\Delta M^{(k)}\big)\rangle \\ + \frac{1}{2}\langle \Xi^\lambda(a, m, \Sigma); \mathrm{Var}_{m,\Sigma}\big(\Delta M^{(k)}\big)\rangle, \end{aligned}$$

where,

$$\begin{aligned} \mathcal{B}^\lambda &:= \sum_k \nu_k^\lambda(a)(\partial_\Sigma f_k), \quad \mathcal{M}^\lambda := \sum_k \nu_k^\lambda(a)(\partial_m f_k), \\ \Xi^\lambda &:= \sum_k \nu_k^\lambda(a)(\partial_m^2 f_k) + \frac{1}{\lambda}\sum_{j,k} \eta_{jk}^\lambda(a)(\partial_m f_j)(\partial_m f_k)^\top, \end{aligned}$$

with $\eta_{jk}^\lambda(a) = \nu_j^\lambda(a)(\mathbb{I}(j=k) - \nu_k^\lambda(a))$. The algorithm thus solves the fixed point equation to find a, and then obtains $\nu^\lambda(a)$ as the randomized action to choose in any timepoint t.

4 Simulation Studies

Dube and Misra [2], in their experiment, randomly assigned one of ten different prices to 7867 different customers who reached Ziprecruiter's paywall. Based on the data, the authors inferred the Θ, and created 5000 independent simulation of markets of one-year length, with $N_t \sim$ Poisson(270). Based on the simulations, they found that the ARC algorithm provides an optimal solution, while the existing methods do not reach the optimal solution even after sufficient trials. Also, ARC and Knowledge Gradient (KG) method achieves the optimal reward with a minimum number of changes in prices and hence is preferable over other algorithms.

5 Conclusion and Comments

1. The authors avoid expressing enough motivation behind the ARC algorithm. However, if the bandits are correlated, and one of the bandits, even being costly, provides extremely crucial information about Θ, ARC algorithm motivates to explore that.

2. There is no clear indication of how the learning function came to be. Intuitively, it uses $\partial_\Sigma f_k$ which denotes the tradeoff between information gain about Θ and the loss of expected reward.

3. The consideration that the probability of buying $p(c_k)$ is the same for every customer is a problem since it depends on particular user preferences.

4. Also, $p(c_k)$ should depend on time t as well, since there is usually a trend effect that motivates investors to buy a particular type of contracts (or on a particular asset) more.

5. More simulations with different incoming distributions other than Poisson should be executed to understand whether the results are consistent. Since the theoretical guarantees only hold asymptotically, a large value of $N_t^{(k)}$ is required to make a reasonable inference.

6. Usually, if c_k is very high, the demand may be extremely low, and in some cases, all $Q_{i,t}^{(k)} = 0$. In such cases, a geometric model could prove useful.

However, the approach presented by the algorithm is novel in the sense that it considers the problem where bandits are correlated with each other, which is of extreme practical importance since, in the dynamic pricing setup described in the paper, common information about a reasonable price range of the product is available to all customers.

References

[1] Samuel N Cohen and Tanut Treetanthiploet. Asymptotic randomised control with applications to bandits. *arXiv preprint arXiv:2010.07252*, 2020.

[2] Jean-Pierre Dubé and Sanjog Misra. Scalable price targeting. Technical report, National Bureau of Economic Research, 2017.

[3] Michael N Katehakis and Arthur F Veinott Jr. The multi-armed bandit problem: decomposition and computation. *Mathematics of Operations Research*, 12(2):262–268, 1987.

[4] Paat Rusmevichientong and John N Tsitsiklis. Linearly parameterized bandits. *Mathematics of Operations Research*, 35(2):395–411, 2010.

[5] Daniel Russo, Benjamin Van Roy, Abbas Kazerouni, Ian Osband, and Zheng Wen. A tutorial on thompson sampling. *arXiv preprint arXiv:1707.02038*, 2017.

[6] William R Thompson. On the likelihood that one unknown probability exceeds another in view of the evidence of two samples. *Biometrika*, 25(3/4):285–294, 1933.

[7] Tanut Treetanthiploet and Samuel N Cohen. Correlated bandits for dynamic pricing via the arc algorithm. *Available at SSRN 3781766*, 2021.

[8] Joannes Vermorel and Mehryar Mohri. Multi-armed bandit algorithms and empirical evaluation. In *European conference on machine learning*, pages 437–448. Springer, 2005.