Multi-Armed Bandits Algorithms for Pricing and Advertising

Marco Mussi

Abstract Nowadays, when it comes to selling a product online, two of the most significant factors are the pricing strategy and the investments in advertising. When determining the price of a product, it is essential to strike a balance. The price should neither be set too low, as this would result in a reduced revenue, nor too high, as it may deter potential buyers. The amount of money we invest in advertising should be balanced to let people know our offer without overspending. These two aspects are usually handled disjointedly by humans, but this may lead to suboptimal solutions. In this work, we focus on the adoption of online learning algorithms to solve the task of finding the optimal price for a product and understand how to advertise it properly. We face various aspects of pricing and advertising, offering theoretical frameworks to address the associated challenges. We start discussing pricing methods, with emphasis on the problem of learning in the presence of temporal dynamics. Then, we discuss the theoretical aspects of advertising, with a particular focus on marketing mix models. Finally, we bring together the problems of pricing and advertising, presenting a unified view.

1 Introduction

Motivated by the rapid increase in the quantity of data and the exponential growth of online platforms, companies are continually seeking innovative strategies to enhance their market presence, capture consumer attention, and optimize their pricing models. Machine Learning (ML) has emerged in recent years as a groundbreaking transformative force, empowering organizations to revolutionize the way they price products and promote them through advertising. The traditional paradigms of pricing and advertising, once reliant on static models and generalized strategies, are rapidly giving way to data-driven, adaptive approaches powered by ML algorithms.

Marco Mussi
Politecnico di Milano, Piazza Leonardo da Vinci, 32, Milan, Italy, e-mail: marco.mussi@polimi.it

In this work, we face the problem of online decision-making in the context of dynamic pricing and advertising budget optimization. These two topics are, indeed, two sides of the same coin. In order to sell a product, we must be able to select both a price that is proper for the reference market and advertise it properly. The price should neither be set too low, as this would result in reduced revenue from the single sale, nor too high, as it may deter potential buyers. The amount of money we invest in advertising should be balanced to let people know of us without overspending and reaching people who are not interested. The goal, indeed, is to optimize the combination of pricing and advertising policies to increase our revenue.

Structure and Contributions. In Section 2, we formulate a new approach (Bacchiocchi et al., 2024) for handling dynamic pricing using Multi-Armed Bandits (MABs, Lattimore and Szepesvári, 2020) methods taking into account the temporal dependencies through the introduction of AutoRegressive (AR) processes to model such a dependency. Such processes are useful to represent trends that are not captured by standard MABs. In Section 3, we focus on the problem of budget optimization in online advertising, and in particular, on the problem of budget optimization in Marketing Mix Models (MMMs). We propose (Mussi et al., 2023), a framework to face the problem of optimizing the budget allocation in MMMs online. In Section 4, we face the problem of jointly optimizing the price at which we want to sell an item and the expenditure to advertise it. We propose (Mussi et al., 2024, 2025), a new framework for handling the problem in which the reward is factored and observable in intermediate steps, and we design an algorithm to solve this problem with theoretical guarantees.[1] These three parts are all binded each other from (*i*) the scope of the proposed algorithms, whose final goal in all the cases is to improve the revenues due to the sales we perform, and (*ii*) the methodology used to pursue the goal, as all the algorithms presented in this work are based on MABs.

2 Dynamic Pricing

In this section, we consider the problem of finding the optimal price for a given product. Our goal is to maximize a certain index, e.g., volumes, turnover, or profit. Usually, pricing algorithms focus on the *one-step* performance (Mussi et al., 2022). These solutions, however, fail in modeling the *long-term* phenomena that a pricing strategy inherently presents. Indeed, with one-step solutions, we fail (*i*) to model the long-term effect such as customer loyalty, and (*ii*) to capture the different demands of loyal and non-loyal customers. This problem, even if ubiquitous in the real world, is unexplored in the literature, as existing approaches struggle to correctly deal with these autoregressive dynamics. Motivated by the problem described above, we propose a novel setting, named *AutoRegressive Bandits* (ARBs), in which the reward follows an AR process of order n whose parameters depend on the actions.

[1] For all the formal proofs and additional results, we refer the interested reader to the works cited above and to (Mussi, 2023).

2.1 Setting

Let $T \in \mathbb{N}$ be the learning horizon. At every round $t \in [\![T]\!]$, the learner chooses an action $a_t \in \mathcal{A} \coloneqq [\![k]\!]$, among the $k \in \mathbb{N}$ available ones. In the ARB setting, the reward evolves according to an *autoregressive process of order n* (AR(n)). Thus, the learner observes a noisy reward x_t of the form:

$$x_t = \gamma_0(a_t) + \sum_{i=1}^{n} \gamma_i(a_t) x_{t-i} + \epsilon_t,$$

where $\gamma_0(a_t) \in \mathbb{R}$ and $(\gamma_i(a_t))_{i \in [\![n]\!]} \in \mathbb{R}^n$ are the unknown *parameters* depending on chosen action a_t, and ϵ_t is σ^2-subgaussian noise. The reward evolution can be also expressed as $x_t = \langle \boldsymbol{\gamma}(a_t), \mathbf{z}_{t-1} \rangle + \epsilon_t$, where $\mathbf{z}_{t-1} \coloneqq (1, x_{t-1}, \dots, x_{t-n})^\intercal \in \mathcal{Z} \coloneqq \{1\} \times \mathcal{X}^n$ is the *vector of past rewards* expressing past history, and $\boldsymbol{\gamma}(a) \coloneqq (\gamma_0(a), \dots, \gamma_n(a))^\intercal \in \mathbb{R}^{n+1}$ is the *parameter vector*, defined for every $a \in \mathcal{A}$. We introduce the following assumptions:

a. (Non-negative coefficients) $\gamma_i(a) \geq 0$ for every $a \in \mathcal{A}$, $i \in [\![0, n]\!]$;
b. (Stability) $\Gamma \coloneqq \max_{a \in \mathcal{A}} \sum_{i=1}^{n} \gamma_i(a) < 1$;
c. (Boundedness) $m \coloneqq \max_{a \in \mathcal{A}} \gamma_0(a) < +\infty$.

The performance of a policy $\boldsymbol{\pi}$ is evaluated in terms of the *expected cumulative reward* over the horizon T, defined as:

$$J(\boldsymbol{\pi}, T) \coloneqq \mathbb{E}\left[\sum_{t=1}^{T} x_t\right].$$

A policy $\boldsymbol{\pi}^*$ is *optimal* if it maximizes the expected average reward, i.e., $\boldsymbol{\pi}^* \in \arg\max_{\boldsymbol{\pi}} J(\boldsymbol{\pi}, T)$. The goal of the learner is to minimize the *expected cumulative (policy) regret* by playing a policy $\boldsymbol{\pi}$, competing against the optimal policy $\boldsymbol{\pi}^*$ over the *learning horizon* T:

$$R(\boldsymbol{\pi}, T) = J(\boldsymbol{\pi}^*, T) - J(\boldsymbol{\pi}, T) = \mathbb{E}\left[\sum_{t=1}^{T} r_t\right],$$

where $r_t \coloneqq x_t^* - x_t$ is the instantaneous policy regret and $(x_t^*)_{t \in [\![T]\!]}$ is the sequence of rewards observed by playing the optimal policy. The optimal policy, which maximizes the expected cumulative reward, is $\boldsymbol{\pi}_t^* \in \arg\max_{a \in \mathcal{A}} \langle \boldsymbol{\gamma}(a), \mathbf{z}_{t-1} \rangle$.

Mapping to Pricing. The pricing problem discussed above can be mapped to the ARB setting. Imagine we want to maximize the volumes over time. The volumes are our reward x_t, and the history of our rewards $x_{t-1}, \dots, x_{t-n}$ provides an indication of the loyal customer pool over the past n units of time (e.g., weeks). The ARB setting allows modeling a reward which is the contribution of both new customers (via γ_0) and the loyal customer pool (via $\gamma_1, \dots, \gamma_n$). Specifically, the price (our *action*), induces different values of the coefficient $\boldsymbol{\gamma}(a_t)$, to represent the different demand curves that loyal and new customers might have.

Algorithm 1: AR-UCB.

Input: Regularization param. λ, AR order n, Exploration coefficients $(\beta_{t-1})_{t\in[\![T]\!]}$
Initialize $\mathbf{V}_0(a) = \lambda\mathbf{I}_{n+1}$, $\mathbf{b}_0(a) = \mathbf{0}_{n+1}$, $\widehat{\boldsymbol{\gamma}}_0(a) = \mathbf{0}_{n+1}$, $\forall a \in \mathcal{A}$, $\mathbf{z}_0 = (1, 0, \dots, 0)^\intercal$, $t \leftarrow 1$
for $t \in [\![T]\!]$ **do**
  Compute $a_t \in \arg\max_{a\in\mathcal{A}} \mathrm{UCB}_t(a) \coloneqq \langle \widehat{\boldsymbol{\gamma}}_{t-1}(a), \mathbf{z}_{t-1}\rangle + \beta_{t-1}(a)\, \|\mathbf{z}_{t-1}\|_{\mathbf{V}_{t-1}(a)^{-1}}$
  Play action a_t and observe $x_t = \langle \boldsymbol{\gamma}(a_t), \mathbf{z}_{t-1}\rangle + \epsilon_t$
  Update $\forall a \in \mathcal{A}$:
    $\mathbf{V}_t(a) = \mathbf{V}_{t-1}(a) + \mathbf{z}_{t-1}\mathbf{z}_{t-1}^\intercal \mathbb{1}_{\{a=a_t\}}$
    $\mathbf{b}_t(a) = \mathbf{b}_{t-1}(a) + \mathbf{z}_{t-1} x_t \mathbb{1}_{\{a=a_t\}}$
    $\widehat{\boldsymbol{\gamma}}_t(a) = \mathbf{V}_t(a)^{-1}\mathbf{b}_t(a)$
  Update $\mathbf{z}_t = (1, x_t, \dots, x_{t-n+1})^\intercal$, $t \leftarrow t+1$
end

2.2 Algorithm

We present AutoRegressive Upper Confidence Bound (AR-UCB, Algorithm 1), an optimistic regret minimization algorithm for the ARB setting. AR-UCB leverages the myopic optimal policy for ARBs and implements an incremental regularized least squares procedure to estimate the unknown parameters $\boldsymbol{\gamma}(a)$, for every action $a \in \mathcal{A}$ independently. The algorithm requires knowledge of the order n of the AR process, although this knowledge can be replaced with that of an upper bound $\overline{n} > n$ of the AR order. AR-UCB starts by initializing for all the actions $a \in \mathcal{A}$ the Gram matrix $\mathbf{V}_0(a) = \lambda\mathbf{I}_{n+1}$, where $\lambda > 0$ is the Ridge regularization parameter, the vectors $\mathbf{b}_0(a) = \widehat{\boldsymbol{\gamma}}_0(a) = \mathbf{0}_{n+1}$, and the observations vector $\mathbf{z}_0 = (1, 0, \dots, 0)^\intercal$. Then, for each round $t \in [\![T]\!]$, AR-UCB computes the *Upper Confidence Bound* (UCB) index for every $a \in \mathcal{A}$ and select the optimistic action a_t as:

$$a_t \in \arg\max_{a\in\mathcal{A}} \ \mathrm{UCB}_t(a) \coloneqq \langle \widehat{\boldsymbol{\gamma}}_{t-1}(a), \mathbf{z}_{t-1}\rangle + \beta_{t-1}(a)\, \|\mathbf{z}_{t-1}\|_{\mathbf{V}_{t-1}(a)^{-1}},$$

where $\widehat{\boldsymbol{\gamma}}_{t-1}(a)$ is the most recent estimate of the parameter vector $\boldsymbol{\gamma}(a)$, $\mathbf{z}_{t-1} = (1, x_{t-1}, \dots, x_{t-n})^\intercal$ is the observations vector, and $\beta_{t-1}(a) > 0$ is a properly selected exploration coefficient. The index $\mathrm{UCB}_t(a)$ is designed to be optimistic, i.e., $\langle \boldsymbol{\gamma}(a), \mathbf{z}_{t-1}\rangle \le \mathrm{UCB}_t(a)$ with high probability for all $a \in \mathcal{A}$. Then, action a_t is executed and the new reward x_t is observed. This sample is employed to update the Gram matrix estimate $\mathbf{V}_t(a_t)$, the vector $\mathbf{b}_t(a_t)$, and the estimate $\widehat{\boldsymbol{\gamma}}_t(a_t)$.

Regret Guarantees. AR-UCB suffers an expected policy regret as follows:

$$\mathbb{E}[R(\texttt{AR-UCB}, T)] \le \widetilde{O}\left(\frac{(m+\sigma)(n+1)^{3/2}\sqrt{kT}}{(1-\Gamma)^2}\right).$$

3 Advertising Optimization

In online advertising, the process that leads to a *conversion* presents complex dynamics and may involve different types of campaigns, and a profitable budget investment policy has to account for their interplay (Court et al., 2009). Indeed, a conversion should be attributed not only to the latest ad and the *joint* consideration of campaigns is fundamental. Consider a simplified model with two types of campaigns: *awareness* (i.e., impression) ads and *conversion* ads. If we evaluate our performance in terms of conversions, we observe that impression ads are not effective, so we will be tempted to reduce their budget. However, this approach may be sub-optimal, as impression ads enhance the effectiveness of conversion ads by increasing the likelihood that users will convert. In addition, the effect of some ads, especially the ones via television, may be delayed, and it has been demonstrated (Chapelle, 2014) that users remember ads in a vanishing way. To model this scenario, we propose *Dynamical Linear Bandits* (DLBs), to model these effects as a linear system with hidden state.

3.1 Setting

In a DLB, we have a *hidden* state $\mathbf{x} \in \mathcal{X}$, where $\mathcal{X} \subseteq \mathbb{R}^n$ is the state space. At each round t, the environment is in the hidden state $\mathbf{x}_t \in \mathcal{X}$, the learner chooses an action $\mathbf{u}_t \in \mathcal{U}$, where $\mathcal{U} \subseteq \mathbb{R}^d$ is the action space. The learner receives a noisy reward $y_t = \langle \boldsymbol{\omega}, \mathbf{x}_t \rangle + \langle \boldsymbol{\theta}, \mathbf{u}_t \rangle + \eta_t$, where $\boldsymbol{\omega} \in \mathbb{R}^n$, $\boldsymbol{\theta} \in \mathbb{R}^d$ are unknown, and η_t is σ^2-subgaussian noise. Then, the environment evolves according to the unknown linear dynamics $\mathbf{x}_{t+1} = \mathbf{A}\mathbf{x}_t + \mathbf{B}\mathbf{u}_t + \boldsymbol{\epsilon}_t$, where $\mathbf{A} \in \mathbb{R}^{n \times n}$ is the dynamic matrix, $\mathbf{B} \in \mathbb{R}^{n \times d}$ is the action-state matrix, and $\boldsymbol{\epsilon}_t$ is a σ^2-subgaussian noise vector. We consider stable systems in which $\mathbf{A}$ has maximum eigenvalues smaller than 1 in module ($\rho(\mathbf{A}) < 1$). Given a policy $\boldsymbol{\pi}$, we define its *(infinite-horizon) expected average reward*:

$$J(\boldsymbol{\pi}) \coloneqq \liminf_{H \to +\infty} \mathbb{E}\left[\frac{1}{H}\sum_{t=1}^{H} y_t\right].$$

A policy $\boldsymbol{\pi}^*$ is an *optimal policy* if it maximizes the expected average reward. We evaluate policies in terms of *expected cumulative regret*, i.e., the sum over time of the difference in performance w.r.t. the optimal policy $\boldsymbol{\pi}^*$.

Lower Bound. We characterize the expected regret that every policy $\boldsymbol{\pi}$ will suffer:

$$\mathbb{E}[R(\boldsymbol{\pi}, T)] \geq \Omega\left(\frac{d\sqrt{T}}{\sqrt{(1 - \rho(\mathbf{A}))}}\right).$$

Mapping to Advertising. Budget allocation in MMMs can be mapped to a DLB where the budget is our action $\mathbf{u}_t$, the value of awareness (not measurable) is the hidden state $\mathbf{x}_t$, and the reward y_t is the number of conversions (observed).

Algorithm 2: `DynLin-UCB`.

Input: Regularization param. λ, Exploration coeffs. $(\beta_{t-1})_{t\in[\![T]\!]}$, Spectral radius UB $\overline{\rho}$
Initialize $t \leftarrow 1$, $\mathbf{V}_0 = \lambda\mathbf{I}_d$, $\mathbf{b}_0 = \mathbf{0}_d$, $\widehat{\mathbf{h}}_0 = \mathbf{0}_d$
Define $M = \min\{M' \in \mathbb{N} : \sum_{m=1}^{M'} 1 + \lfloor \frac{\log m}{\log(1/\overline{\rho})} \rfloor > T\} - 1$
for $m \in [\![M]\!]$ **do**
  Compute $\mathbf{u}_t \in \arg\max_{\mathbf{u}\in\mathcal{U}} \mathrm{UCB}_t(\mathbf{u})$ where $\mathrm{UCB}_t(\mathbf{u}) \coloneqq \langle \widehat{\mathbf{h}}_{t-1}, \mathbf{u}\rangle + \beta_{t-1}\|\mathbf{u}\|_{\mathbf{V}_{t-1}^{-1}}$
  Play arm $\mathbf{u}_t$ and observe reward y_t
  Define $H_m = \lfloor \frac{\log m}{\log(1/\overline{\rho})} \rfloor$
  for $j \in [\![H_m]\!]$ **do**
    Play arm $\mathbf{u}_t = \mathbf{u}_{t-1}$ and observe y_t
    Update $\mathbf{V}_t = \mathbf{V}_{t-1},\ \mathbf{b}_t = \mathbf{b}_{t-1},\ t \leftarrow t+1$
  end
  Update and compute: $\mathbf{V}_t = \mathbf{V}_{t-1} + \mathbf{u}_t\mathbf{u}_t^\mathrm{T},\ \mathbf{b}_t = \mathbf{b}_{t-1} + \mathbf{u}_t y_t,\ \widehat{\mathbf{h}}_t = \mathbf{V}_t^{-1}\mathbf{b}_t,\ t \leftarrow t+1$
end

3.2 Algorithm

We present an *optimistic* regret minimization algorithm for the DLBs setting. Dynamical Linear Upper Confidence Bound (`DynLin-UCB`, Algorithm 2) requires the knowledge of an upper-bound $\overline{\rho} < 1$ on the spectral radius of $\mathbf{A}$. To assess the quality of action $\mathbf{u} \in \mathcal{U}$, we *persist* in applying it so that the system approximately reaches the corresponding steady state and, then, observe the reward y_t, representing a reliable estimate of $J(\mathbf{u}) = \langle \mathbf{h}, \mathbf{u}\rangle$, where $\mathbf{h} = \boldsymbol{\theta} + \mathbf{B}^\mathrm{T}(\mathbf{I}_n - \mathbf{A})^{-\mathrm{T}}\boldsymbol{\omega}$ is what we call a *Markovian* vector representing the whole system at the steady state. We shall show that the number of rounds needed to approximately reach such a steady state is logarithmic in the learning horizon T and depends on $\overline{\rho}$. `DynLin-UCB` subdivides the learning horizon T into M *epochs*. Each epoch $m \in [\![M]\!]$ is composed of $H_m + 1$ rounds, where $H_m = \lfloor \log m / \log(1/\overline{\rho}) \rfloor$. At the beginning of each epoch, `DynLin-UCB` computes the UCB index defined for every $\mathbf{u} \in \mathcal{U}$ as $\mathrm{UCB}_t(\mathbf{u}) \coloneqq \langle \widehat{\mathbf{h}}_{t-1}, \mathbf{u}\rangle + \beta_{t-1}\|\mathbf{u}\|_{\mathbf{V}_{t-1}^{-1}}$, where $\widehat{\mathbf{h}}_{t-1} = \mathbf{V}_{t-1}^{-1}\mathbf{b}_{t-1}$ is the Ridge regression estimator of a Markov parameter $\mathbf{h}$, and $\beta_{t-1} > 0$ is a properly selected exploration coefficient. Similar to `Lin-UCB` (Abbasi-Yadkori et al., 2011), the index $\mathrm{UCB}_t(\mathbf{u})$ is designed to be optimistic, i.e., $J(\mathbf{u}) \le \mathrm{UCB}_t(\mathbf{u})$ in high-probability for all $\mathbf{u} \in \mathcal{U}$. The optimistic action $\mathbf{u}_t \in \arg\max_{\mathbf{u}\in\mathcal{U}} \mathrm{UCB}_t(\mathbf{u})$ is executed and persisted for the next H_m rounds. In this way, at the end of each epoch, the reward y_t is an almost-unbiased sample of the steady-state performance $J(\mathbf{u}_t)$, which can be employed to update the $\mathbf{V}_t$ and $\mathbf{b}_t$.

Regret Guarantees. Considering a proper selection of β_t and the knowledge of the upper bounds $\overline{\rho} < 1$, `DynLin-UCB` suffers an expected regret bounded as:

$$\mathbb{E}[R(\texttt{DynLin-UCB}, T)] \le \widetilde{\mathcal{O}}\left(\frac{d\sqrt{T}}{1-\overline{\rho}} + \frac{\sqrt{dT}}{(1-\overline{\rho})^{3/2}} + \frac{1}{(1-\rho(\mathbf{A}))^2}\right).$$

4 Joint Pricing and Advertising

In this section, we present a model for jointly optimizing pricing and advertising. We have to coherently choose (*i*) the *price* and (*ii*) how much *budget* to invest in advertising. The price we set determines the willingness of the users to buy a given item, i.e., the *conversion rate*, while the advertising budget influences the number of people that will see an item, i.e., the number of *impressions*. At every step, we select a *price-budget* couple, and we observe the *conversion rate*, which depends on the price, and the number of *impressions*, which depends on the *budget* we invest in advertising. This scenario can be treated as a standard MAB by looking just at the reward (i.e., the revenue) and considering price-budget couples as actions. However, this solution is very inefficient, and the resulting problem will present an unnecessarily large action space, including all the possible combinations of actions. Given that, we now propose a general model, called *Factored Reward Bandits* (FRBs), able to characterize the problem of optimizing this scenario.

4.1 Setting

Let $T \in \mathbb{N}$ be the time horizon. In a FRB, at every round $t \in [\![T]\!]$ we choose an action vector $\mathbf{a}(t) = (a_1(t), \dots, a_d(t))$ in a given action space $\mathcal{A} \coloneqq [\![k_1]\!] \times \cdots \times [\![k_d]\!]$, where $k_i \in \mathbb{N}$ is the number of options for the i^{th} action component, and $d \in \mathbb{N}$ is the action vector dimension (i.e., the number of components that the learner must select). As a result, we observe a vector of d components $\mathbf{x}(t) = (x_1(t), \dots, x_d(t))$. The i^{th} component $x_i(t)$ of the observation vector $\mathbf{x}(t)$ is the effect of the i^{th} action component $a_i(t)$ in the action vector $\mathbf{a}(t)$. Every component of the observation vector $\mathbf{x}(t)$ is independent of the others and sampled from a distribution $x_i(t) \sim \nu_{i,a_i(t)}$. We consider stochastic observations, i.e., $x_i(t) = \mu_{i,a_i(t)} + \epsilon_i(t)$, where $\mu_{i,a_i(t)}$ is the expected value of the observation of action a_i of the i^{th} component, and $\epsilon_i(t)$ is σ^2-subgaussian noise. We consider bounded expected values for the observations, i.e., $\mu_{i,a_i} \in [0,1]$ for every $i \in [\![d]\!]$, $a_i \in [\![k_i]\!]$. The reward is given by the product of the observations $r(t) = \prod_{i \in [\![d]\!]} x_i(t)$. In the FRB setting, the optimal action is:

$$\mathbf{a}^* = (a_1^*, \dots, a_d^*) \in \underset{\mathbf{a}=(a_1,\dots,a_d)\in\mathcal{A}}{\arg\max} \prod_{i \in [\![d]\!]} \mu_{i,a_i},$$

and we can factorize the learning problem observing that $a_i^* \in \arg\max_{a_i \in [\![k_i]\!]} \mu_{i,a_i}$ for every $i \in [\![d]\!]$. We call $\mu_i^* = \mu_{i,a_i^*}$ the expected value of the optimal action of the i^{th} component. Given a policy $\boldsymbol{\pi}$, we define its *cumulative regret* as:

$$R(\boldsymbol{\pi}, T) \coloneqq T \prod_{i \in [\![d]\!]} \mu_i^* - \sum_{t \in [\![T]\!]} \prod_{i \in [\![d]\!]} \mu_{i,a_i(t)}.$$

The goal of the learner is to minimize the *expected cumulative regret* $\mathbb{E}[R(\boldsymbol{\pi}, T)]$.

Algorithm 3: F-UCB.

Input: Exploration param. α, Subgaussianity proxy σ, Action space dim. $k_i, \forall i \in [\![d]\!]$
Initialize $\forall a_i \in [\![k_i]\!],\ i \in [\![d]\!]: \quad N_{i,a_i}(0) \leftarrow 0,\ \widehat{\mu}_{i,a_i}(0) \leftarrow 0$
for $t \in [\![T]\!]$ **do**
  Select $\mathbf{a}(t) \in \arg\max_{\mathbf{a}=(a_1,\dots a_d)\in\mathcal{A}} \prod_{i\in[\![d]\!]} \mathrm{UCB}_{i,a_i}(t)$
  Play $\mathbf{a}(t)$ and observe $\mathbf{x}(t) = (x_1(t), \dots, x_d(t))$
  Update $\forall i \in [\![d]\!]$:
    $\widehat{\mu}_{i,a_i(t)}(t) \leftarrow \frac{\widehat{\mu}_{i,a_i(t)}(t-1)\ N_{i,a_i(t)}(t-1)+x_i(t)}{N_{i,a_i(t)}(t-1)+1},\ \ N_{i,a_i(t)}(t) \leftarrow N_{i,a_i(t)}(t-1)+1$
  Update $\forall i \in [\![d]\!], \forall j \in [\![k_i]\!] \setminus \{a_i(t)\}: \widehat{\mu}_{i,j}(t) \leftarrow \widehat{\mu}_{i,j}(t-1),\ N_{i,j}(t) \leftarrow N_{i,j}(t-1)$
end

Lower Bound. We characterize the expected regret that every policy $\boldsymbol{\pi}$ will suffer:

$$\mathbb{E}[R(\boldsymbol{\pi}, T)] \geq \Omega\left(\sqrt{T \sum_{i\in[\![d]\!]} k_i}\right).$$

4.2 Algorithm

We present an *optimistic* regret minimization algorithm for the FRB setting. Factored Upper Confidence Bound (F-UCB, Algorithm 3) is inspired by the optimistic bound of UCB1 (Auer et al., 2002; Bubeck, 2010). The algorithm requires as input the action space dimension k_i for every $i \in [\![d]\!]$, the exploration parameter α, and the subgaussianity coefficient σ. For every round $t \in [\![T]\!]$, we estimate the best optimistic action, i.e., the action $\mathbf{a}(t)$ maximizing the index:

$$\mathbf{a}(t) \in \underset{\mathbf{a}=(a_1,\dots,a_d)\in\mathcal{A}}{\arg\max} \prod_{i\in[\![d]\!]} \mathrm{UCB}_{i,a_i}(t),$$

where $\mathrm{UCB}_{i,a_i}(t) \coloneqq \widehat{\mu}_{i,a_i}(t-1) + \sigma\sqrt{\frac{\alpha \log t}{N_{i,a_i}(t-1)}}$, calling $\widehat{\mu}_{i,a_i}(t)$ is the empirical mean of the observations for the i^{th} component of the observation vector determined by the action component a_i, and $N_{i,a_i}(t)$ is the number of times the such a component has been played. Once we selected the best action according to our optimistic criterion, we play it and retrieve the observation vector $\mathbf{x}(t) = (x_1(t), \dots, x_d(t))$. We use the observation vector to update the estimators and the related counters.

Regret Guarantees. F-UCB presents a worst-case upper bound as follows:

$$\mathbb{E}[R(\text{F-UCB}, T)] \leq \widetilde{O}\left(\sigma \sum_{i\in[\![d]\!]} \sqrt{T k_i}\right).$$

5 Conclusions

We presented three MAB settings for dynamic pricing and advertising budget optimization. In Section 2, we proposed a model that allows us to model temporal dependencies in pricing through AR processes. In Section 3, we faced an advertising problem, and we focused on a new model to optimize MMMs. In Section 4, we proposed a model to optimize pricing and advertising coherently. For every scenario, we presented an algorithm to handle it, and we discussed its theoretical guarantees.

References

Francesco Bacchiocchi, Gianmarco Genalti, Davide Maran, Marco Mussi, Marcello Restelli, Nicola Gatti, and Alberto Maria Metelli. Autoregressive bandits. In *International Conference on Artificial Intelligence and Statistics (AISTATS)*, 2024.

Tor Lattimore and Csaba Szepesvári. *Bandit algorithms*. Cambridge University Press, 2020.

Marco Mussi, Alberto Maria Metelli, and Marcello Restelli. Dynamical linear bandits. In *International Conference on Machine Learning (ICML)*, 2023.

Marco Mussi, Simone Drago, Marcello Restelli, and Alberto Maria Metelli. Factored-reward bandits with intermediate observations. In *International Conference on Machine Learning (ICML)*, 2024.

Marco Mussi, Simone Drago, Marcello Restelli, and Alberto Maria Metelli. Factored-reward bandits with intermediate observations: Regret minimization and best arm identification. *Artificial Intelligence*, 2025.

Marco Mussi. *Online Learning Methods for Pricing and Advertising*. PhD thesis, Politecnico di Milano, 2023.

Marco Mussi, Gianmarco Genalti, Francesco Trovò, Alessandro Nuara, Nicola Gatti, and Marcello Restelli. Pricing the long tail by explainable product aggregation and monotonic bandits. In *ACM Conference on Knowledge Discovery and Data Mining (KDD)*, 2022.

David Court, Dave Elzinga, Susan Mulder, and Ole Jørgen Vetvik. The consumer decision journey. *McKinsey Quarterly*, 2009.

Olivier Chapelle. Modeling delayed feedback in display advertising. In *ACM Conference on Knowledge Discovery and Data Mining (KDD)*, 2014.

Yasin Abbasi-Yadkori, Dávid Pál, and Csaba Szepesvári. Improved algorithms for linear stochastic bandits. In *Advances in Neural Information Processing Systems (NIPS)*, 2011.

Peter Auer, Nicolo Cesa-Bianchi, and Paul Fischer. Finite-time analysis of the multiarmed bandit problem. *Machine Learning*, 2002.

Sébastien Bubeck. *Bandits games and clustering foundations*. PhD thesis, Université des Sciences et Technologie de Lille, 2010.