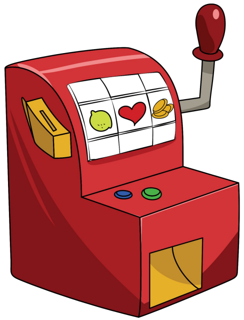
Extension of Asymptotic Randomized Control Algorithm

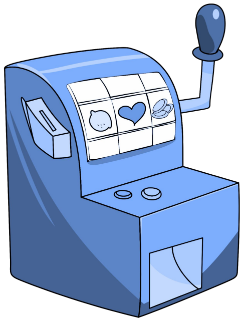
Subhrajyoty Roy (MB1911)

Paper: “Correlated Bandits for Dynamic Pricing via the ARC Algorithm”
Date: 16th May, 2021

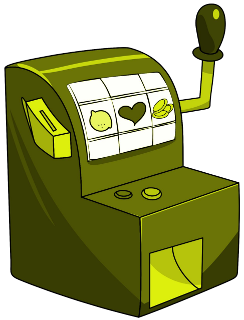
Multi-armed Bandit Problem



Reward ~ F_1



Reward ~ F_2



Reward ~ F_3

Exploration vs Exploitation

Example: Dynamic Pricing

- A store with an item.
- Choice of possible prices $\{p_1, p_2, \ldots, p_k\}$
- Demand is an unknown decreasing function of the price.
- The number customer who buys

$$Y_t \sim \mathrm{Bin}(N_t, \theta(p_t))$$

- If low price, more N_t, more accurate inference. Less reward.
- If high price, less reliable estimate, more reward.

Asymptotic Randomized Control Algorithm

Assumption: The latent parameter θ denoting public preferences are static

$$\text{Objective} = \sum_{t=1}^{\infty} \beta^{(t-1)} \mathbb{E}_{a_t}(R(t, a_t) \mid \mathcal{F}_{t-1})$$

Optimal action satisfies $a_t = a(\mathcal{F}_{t-1})$ and hence a fixed point equation $a = G(a)$

Direction of extension

Modified Setup:

$$
\begin{aligned}
X_{t+1} &= A_{a_t} + BX_t + w_t, \qquad w_t \sim N(0, \Sigma_w) \\
\mathrm{logit}(\theta_t(a_t)) &= \alpha + \beta a_t + \Gamma X_t + v_t, \qquad v_t \sim N(0, \sigma_v^2) \\
Y_{a_t,t} &\sim \mathrm{Bin}(N_t, \theta_t(a_t)) \\
R_t(a_t) &= a_t Y_{a_t,t}
\end{aligned}
$$

Modified Objective:

$$
V(a_1, \ldots a_T) = \sum_{t=1}^{T} \beta^{(t-1)} \mathbb{E}(R_t(Y_t(a_t)) \mid \mathcal{F}_{t-1}), \qquad \text{where } \beta \in (0,1)
$$

Extension approach

Initial Rounds
More exploration
Less exploitation

Final Rounds
Less exploration
More exploitation

$$a_T = \arg\max_{a \in \mathcal{A}} \mathbb{E}(R_T(Y_T(a)) \mid \mathcal{F}_{T-1})$$

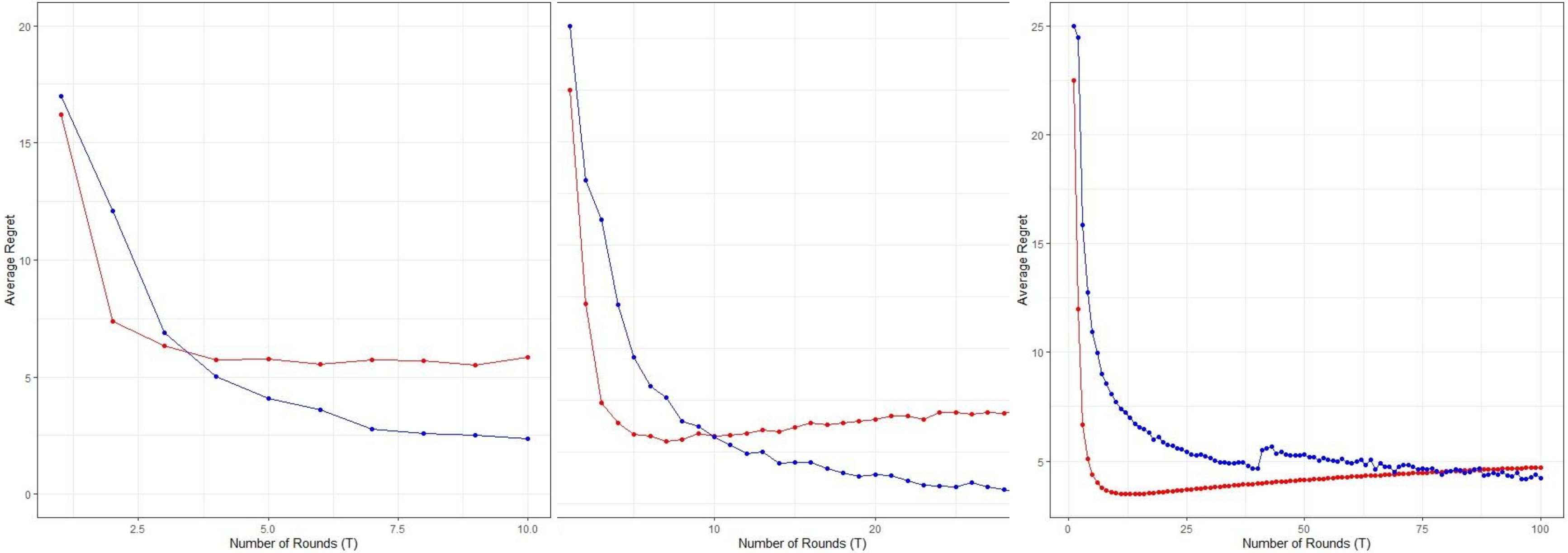
$$a_{T-1} = \arg\max_{a \in \mathcal{A}} \left[\mathbb{E}(R_{T-1}(Y_{T-1}(a)) \mid \mathcal{F}_{T-2}) + \beta \max_{b \in \mathcal{A}} \mathbb{E}(R_T(Y_T(b)) \mid \mathcal{F}_{T-2} \cup \{a_{T-1} = a\}) \right]$$

Possible to use Taylor’s theorem to approximate the 2nd term using solution at T-th round.
Hence, a backward induction approach.

Simulation

Number of rounds vs Average Regret (=1/T * (maximal total reward - current total reward)

T = 10 T = 30 T = 100



Pros and Cons

- Enables practical short term profitability.
- Allows dynamic evolution of the latent variables like public preference.
- Better than ARC algorithm in last few rounds.
- Backward induction requires future values of be computed, which are needed to be stored. Hence a memory constraint.
- Need to change optimal action too frequently. May be inconvenient and costly.

THANK YOU