

Retail Price Optimisation through Multi-Armed Bandit Experiments

V Shekar
Sr. Data Scientist, PPCI
Lowe's Services India Pvt Ltd
Bengaluru, India
v.shekar@lowes.com

Abstract — Optimizing the prices of products in the retail landscape while carrying a wide range of products remains a challenge. The breadth of the product catalogue, varying demand behaviours across categories, stock availability, competitiveness, changes in macroenvironment, changes in demand patterns, or even the absence of demand patterns elevate the complexity of pricing products. A near-optimal price will always reap benefits for both retailers and their customers. Selling items at deep discounts would negatively impact profit margins, and pricing items higher would impact revenue and customer satisfaction. In order to successfully find an optimal price to balance both profitability and revenue, price optimisation is vital. Often, optimising the price of a single item would require business knowledge, understanding its historical price sensitivity, customers willingness to pay, and many other factors. Due to this, many retailers price items manually. But very few test the price to observe sensitivity or elasticity to arrive at a pricing decision. Test-learn based price decisions have proven to identify optimal prices when done in real time. Frameworks like Multi-Armed Bandit Experiments help in testing multiple price points simultaneously and learning from the demand response through a finite number of actions. Also, with innovation in competitive intelligence and the ability for online retail channels to more quickly see and respond to demand responses, the useful life of an optimal price for the key value items in an industry is much shorter and sometimes an optimal price may need to be determined intraday. Therefore, use of this reinforcement learning framework at scale, the simplicity of real-time implementation to identify the current demand pattern for an item, helping in deciding the optimal price, are explained in this paper.

Keywords — Price Optimization, Price Testing, Home Improvement Retail, Multi-Armed Bandit Experiments, Reinforcement Learning.

I. INTRODUCTION

The retail industry has undergone a transformative revolution in recent years, largely driven by advancements in technology. One of the critical aspects of retail operations that technology has significantly impacted is pricing decisions. In the current generation, retailers are increasingly relying on sophisticated data analytics, artificial intelligence (AI), and machine learning algorithms to determine and optimize pricing strategies.

A. Role of Technology in Pricing Decisions

1. **Data-Driven Insights:** Technology has enabled retailers to collect and analyse vast amounts of data from various sources, including customer

behaviour, market trends, and competitor pricing. This data-driven approach allows retailers to make informed pricing decisions that are more aligned with consumer preferences and market conditions.

2. **Dynamic Pricing:** Dynamic pricing, made possible by real-time data analysis, allows retailers to adjust prices based on supply and demand fluctuations, seasonal changes, and even individual customer behaviour. This dynamic approach maximizes revenue while ensuring competitiveness.
3. **Competitor Analysis:** Advanced algorithms can continuously monitor competitors' pricing strategies and adjust a retailer's prices accordingly. This ensures that retailers remain competitive without the need for manual price checking.
4. **Personalization:** Technology enables retailers to create personalized pricing strategies for individual customers. By analysing past purchase history and behaviour, retailers can offer tailored discounts and promotions, increasing customer loyalty.
5. **Optimization:** Machine learning algorithms can optimize pricing strategies by testing various pricing points and evaluating their impact on sales and profitability. This helps retailers find the ideal balance between price and revenue.

B. Benefits of Technology-Driven Pricing

1. **Increased Profitability:** Technology-driven pricing strategies maximize profitability by ensuring that prices are set at levels that customers are willing to pay while also considering costs and market conditions.
2. **Competitive Advantage:** Retailers using advanced pricing technology can respond quickly to market changes and competitor moves, giving them a competitive edge in a fast-paced industry.
3. **Enhanced Customer Experience:** Personalized pricing can improve the shopping experience, making customers feel valued and more likely to return for future purchases.

4. Efficiency: Automated pricing decisions reduce the need for manual price adjustments, saving time and resources for retailers.

Technology-driven pricing decisions have become a cornerstone of the modern retail industry. Leveraging data analytics, AI, and machine learning, retailers can optimize prices, enhance profitability, and deliver a more personalized shopping experience. In the ever-evolving retail landscape, embracing technology in pricing decisions is not just an option; it's a necessity for staying competitive and meeting the expectations of the current generation of consumers. This paper explains the efficient and scalable process of optimizing prices by identifying the demand patterns of products and uncovering price distribution the masses are willing to pay.

II. BACKGROUND - CONCEPTS

Multi-armed bandit (MAB) experiments are a class of online decision-making problems that originated in the field of statistics and have found applications in various domains, including machine learning, marketing, healthcare, and more. The name "multi-armed bandit" is derived from the idea of a gambler facing multiple slot machines (the "bandits") and trying to maximize their total reward by deciding which machines to play.

In a typical MAB setup, it has a set of arms (options or actions), each associated with an unknown reward distribution. The goal is to iteratively choose arms to maximize the cumulative reward over time while learning about the rewards associated with each arm. The challenge is to balance exploration (trying different arms to gather information) and exploitation (choosing the arm that appears to be the best based on the current knowledge).

A. Components of a Multi-Armed Bandit Experiment

1. Arms: These represent the different options or actions it can take. For example, in a marketing context, arms could represent different ad campaigns or strategies.
2. Reward Distributions: Each arm is associated with a probability distribution from which rewards are generated. The goal is to estimate the mean or expected reward of each arm.
3. Exploration vs. Exploitation: This is the core trade-off in MAB problems. It needs to decide how often to explore (try different arms) to gather information and when to exploit (choose the arm with the highest estimated reward) to maximize rewards.

B. Components of a Multi-Armed Bandit Experiment:

1. Online Advertising: Determine which ad creative, campaign, or bidding strategy to use to maximize click-through rates or conversions.

2. Clinical Trials: Optimize the allocation of patients to different treatments in medical trials to identify the most effective treatment faster.
3. Content Recommendation: Choose which articles, videos, or products to recommend to users on platforms like Netflix or Amazon.
4. A/B Testing: Optimize A/B tests by dynamically allocating traffic to different variants based on ongoing performance.
5. Dynamic Pricing: Adjust pricing in real-time to maximize revenue, especially in e-commerce and ride-sharing services.

Overall, multi-armed bandit experiments have gained popularity due to their ability to balance exploration and exploitation effectively, leading to improved decision-making and better outcomes in various domains. These experiments continue to be an active area of research, with ongoing developments and refinements in algorithm design and application.

III. METHODOLOGY

In the absence of a known demand function, optimising price of product would require real time demand knowledge for a range of prices. Multi-Armed Bandit experiments allows to learn the demand function while testing different prices dynamically. The dynamic price optimisation problem is formulated by defining the range of price to test. In order to set a range of price and have finite number (K) of prices, setting lower (p_L) and upper (p_U) price limit according to pricing strategies becomes important. A margin increase strategy would always want to test for higher than current price, and a sell through maximization strategy would want to test lower than current price. Once the lower & upper price limits are set, finite K prices such that $p_L < \{p_1, p_2, \dots, p_k\} < p_U$. This would lead us to build a K-armed bandit that learn demand information $D(p)$ of each price.

Depending on the maximization strategy, objective function can be defined by deriving from demand $D(p)$. Revenue realization at price p is $R(p) = p * D(p)$ and Profit Margin being $M(p) = (p-c) * D(p)$ where c is cost of product for demand $D(p)$. A weighted objective function F can be defined by giving weightage to revenue $R(p)$, profit margin $M(p)$ and demand $D(p)$ so that one function can be reused for different strategies.

$$F = w_r * R(p) + w_m * M(p) + w_d * D(p) \text{ where } w_r + w_m + w_d = 1$$

K-armed bandit acts as an agent executing k prices in real world environment and evaluates the reward of a price which is also known as regret as per weighted objective function $F(p)$. The K-armed bandit setup consists of 2 phases for testing & learning known as exploration & exploitation phases. During exploration phase, the setup tests prices and records the regret for each of the price. The gained knowledge is utilized during exploitation phase to test the best price so to maximize the regret or increase the reward earned. The

UCB1-Tuned algorithm, a popular method is incorporated for solving this K-armed bandit.

The UCB1-Tuned algorithm is an extension of the UCB1 (Upper Confidence Bound 1). UCB1-Tuned, like its predecessor, is designed to balance exploration and exploitation to maximize cumulative reward while making decisions in a sequential manner. However, it incorporates a "tuning" mechanism that adjusts the exploration-exploitation trade-off based on observed outcomes.

UCB1-Tuned maintains estimates of the mean reward for each price (arm) and the number of times each price was executed. At each time step, it selects the price (arm) with the highest Upper Confidence Bound, which combines the mean reward estimate and a measure of uncertainty. The tuned UCB value for a price (arm) p at time t is calculated as:

$$\text{UCB1-Tuned}(p, t) = \text{average}(F(p)) + \sqrt{(\log(t) / \text{num_pulls}(p)) * \min(\text{var}(F(p)), 0.25)}$$

Where:

- $\text{average}(F(p))$ is estimated mean of weighted objective function $F(p)$ for price p .
- $\text{var}(F(p))$ is estimated variance of weighted objective function $F(p)$ for price p .
- t represents total number of times all k prices were executed
- $\text{num_pulls}(p)$ are total number of times price p was executed.

The UCB1-Tuned algorithm uses these estimates to compute a "tuned" UCB value for each price (arm). The key difference is that UCB1-Tuned uses the variance estimate to scale the exploration term dynamically. This means arms with high uncertainty (high variance) are explored more aggressively. By incorporating the variance estimate and the tuning parameter, UCB1-Tuned adjusts its exploration strategy as it gains more information about the price reward distributions. Prices that are more uncertain are explored further, while prices with clearer reward estimates are exploited more aggressively. UCB1-Tuned has been shown to provide improved regret bounds (i.e., how much cumulative regret the algorithm incurs compared to an optimal strategy) compared to the basic UCB1 algorithm in certain scenarios, making it a valuable choice for applications where balancing exploration and exploitation is critical.

IV. SIMULATION RESULTS

Assuming a product costs \$100, setting (p_L, p_U) to (\$125, \$185), 11 prices were selected between (p_L, p_U) in increment of \$5 where $p \in \{\$130, \$135, \$140, \$145, \$150, \$155, \$160, \$165, \$170, \$175, \$180\}$. A customer's willingness to pay (WTP) according to desired strategy was discovered by demand responses for each price points. UCB1-Tuned algorithm was simulated on 11-armed bandit with only 4 arms allowed to execute an action at any time t . Number of trials

are set to 56 (days) with 4 prices (arms) executed each day which will lead to total of 224 times all prices are executed.

Customers were sampled from a normal distribution with mean WTP of \$165. Maximization goal was to increase revenue, profit margin & units sold with weightage of 80%, 15% and 5% respectively. Such that $F(p) = 0.80 * R(p) + 0.15 * M(p) + 0.05 * D(p)$.

Post 56 trials of executing 4 prices each trial, $F(p)$ was maximized at $p = \$155$ indicating optimal price as \$155 where most customers would pay for the product so that set objective goal is achieved.

TABLE I. MAB PRICE SIMULATION

Price	# of Trials	$F(p)$ per Trial
\$130	15	404
\$135	18	459
\$140	21	495
\$145	24	535
\$150	17	541
\$155	19	560
\$160	21	494
\$165	32	365
\$170	25	206
\$175	19	77
\$180	13	98

V. CONCLUSION

Multi-armed bandit algorithms excel at making decisions in real-time. In dynamic pricing and price personalization, market conditions, customer preferences, and competitive landscapes can change rapidly. Multi-armed bandits allow to adapt your pricing strategy on the fly to optimize outcomes.

Multi-armed bandits can be applied to personalized pricing. By segmenting customers based on their behavior, preferences, or demographics, each segment can be treated as a "bandit arm." The algorithm can then allocate resources to determine the optimal price for each segment, tailoring the pricing experience to individual customers.

However, it's essential to note that the effectiveness of multi-armed bandits in dynamic pricing and price personalization depends on factors like the algorithm used, the quality and quantity of data, and the specific business context. Additionally, ethical considerations should be considered when implementing personalized pricing strategies.

REFERENCES

- [1] Kanishka Misra, Eric M. Schwartz, and Jacob Abernethy, "Dynamic Online Pricing with Incomplete Information Using Multi-Armed Bandit Experiments" February 2018.
- [2] Omar Besbes, and Assaf Zeevi, "Dynamic Pricing Without Knowing the Demand Function: Risk Bounds and Near-Optimal Algorithms" Vol. 57, No. 6, November–December 2009, pp. 1407–1420 .