Price Investment using Prescriptive Analytics and Optimization in Retail

Prakhar Mehrotra*
Prakhar.Mehrotra@walmart.com
Walmart Labs
Sunnyvale, CA

Linsey Pang*
Linsey.Pang@walmartlabs.com
Walmart Labs
Sunnyvale, CA

Karthick Gopalswamy*
Karthick.Gopalswamy@walmart.com
Walmart Labs
Bentonville, AR

Avinash Thangali*
Avinash.Thangali@walmartlabs.com
Walmart Labs
Sunnyvale, CA

Timothy Winters
Walmart Labs
Bentonville, AR

Ketki Gupte
Walmart Labs
India

Dnyanesh Kulkarni
Walmart Labs
Bentonville, AR

Sunil Potnuru
Walmart Labs
India

Supreeth Shastry
Walmart Labs
Bentonville, AR

Harshada Vuyyuri
Walmart Labs
Bentonville, AR

ABSTRACT

As the world's largest retailer, Walmart's core mission is to save people money so they can live better. We call the strategy we use to accomplish this goal our *Every Day Low Price* strategy. By keeping operational expenses as low as possible, we can continually apply a downward pressure on our prices, in turn increasing the amount of traffic, and ultimately, sales within our stores. In this paper, we apply Machine Learning (ML) algorithms and Operations Research techniques for forecasting and optimization to build a new price recommendation system, which improves our ability to generate price recommendations accurately and automatically. Comprised of a demand forecasting step, two optimizations, and causal inference analysis, our system was evaluated in the form of forecast backtests and live pricing experiments, both of which suggested that our approach was more effective than the current rule-based pricing system.

ACM Reference Format:
Prakhar Mehrotra, Linsey Pang, Karthick Gopalswamy, Avinash Thangali, Timothy Winters, Ketki Gupte, Dnyanesh Kulkarni, Sunil Potnuru, Supreeth Shastry, and Harshada Vuyyuri. 2020. Price Investment using Prescriptive Analytics and Optimization in Retail. In *Proceedings of the 26th ACM SIGKDD Conference on Knowledge Discovery and Data Mining (KDD '20), August 23–27, 2020, Virtual Event, CA, USA.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3394486.3403365

*These are corresponding authors.

KDD '20, August 23–27, 2020, Virtual Event, CA, USA

ACM ISBN 978-1-4503-7998-4/20/08…$15.00
https://doi.org/10.1145/3394486.3403365

1 INTRODUCTION

Although the rapid expansion of e-commerce platforms has transformed the retail sector, it has not meant the end of traditional brick-and-mortar altogether. Indeed, across a variety of industries, certain brick-and-mortar retailers have not only avoided losing business to e-commerce platforms in recent years, but have managed to thrive in this new environment, particularly by leveraging the advantages traditional retail stores have over e-commerce platforms. Not only does consumer demand for brick-and-mortar shopping experiences remain strong in the United States, but a large physical footprint also provides a retailer with a strong infrastructure to facilitate various efforts like marketing and distribution more easily and effectively than is possible for e-commerce retailers, particularly at a local community level [1]. These developments have demonstrated the necessity for brick-and-mortar retailers to make use of such assets in order to remain competitive. One such resource, however, that has seen limited utilization until recently, is the large amounts of data available to many brick-and-mortar retailers; using such data to inform business decisions has become an area of increased attention and importance within the brick-and-mortar retail sector.

Two of the most consequential decisions retailers must make are (1) determining which products in which to *invest* their available budgets in order to carry them in our stores, and (2) determining what prices at which we should sell them. Precisely how these decisions are made depends on the desired business outcome, as well as a number of constraints imposed by internal rules, implementation feasibility, economic realities, etc. Walmart's process of making these two decisions, called *Price Investment*, occurs at a high level, involving the selection of firstly the best portfolio of *products*, equivalently called *UPCs* or *items*, in which to invest, and then of the best pricing of the various component products of such a portfolio, in order to achieve certain business goals such as driving traffic, improving customer retention, etc. We refer to this process as an *investment* because of our fundamental approach to pricing the items we sell. Our general framework for pricing is determined by our company-wide *Every Day Low Price* (EDLP) business strategy, by which we aim to increase our revenues principally by keeping our prices low in order to increase item demand and traffic to our stores. Referring to our process as an investment reflects the fact that, because of our EDLP strategy, Walmart strictly adjusts prices with reduction, effectively amounting to an additional investment of our total spending budget. Short-term promotional price discounts have been an effective and predominant strategy for many large scale brick-and-mortar retailers to improve performance with respect to long-term business goals such as increasing store traffic and revenues [2]. Walmart, by contrast, in keeping with our EDLP strategy, rather than using such temporary promotions to achieve its business goals, instead focuses on reducing the everyday prices of our products, building a brand reputation of low prices and the highest dollar-value for our customers.

The importance of low everyday prices to our business strategy makes Price Investment a key problem for Walmart to solve correctly. Starting from 2015, Walmart's Central Pricing team has been implementing a rule-based Price Investment strategy to our grocery items, and had by the end of 2018, already deployed this system in over half of our stores. However, this investment framework and associated pricing system have remained largely unchanged. The use of quantitative techniques from the Machine Learning and Operations Research disciplines has proven effective in a variety of business contexts, allowing for both scalability and automation of decision-making over manual alternatives without sacrificing, or even improving upon, reliability. To improve our current pricing strategy in these respects, we introduce a novel approach using Machine Learning and Operations Research techniques to solve our Price Investment task. We call this new Price Investment framework our *Price Recommendation System*, or PRS.

1.1 Business Problem Statement

Our business target, in brief, is to solve our Price Investment task in such a way to maximize the demand, or *demand-volume* in units, for each of our products. Henceforth we refer to increases in demand-volume as *lift*.

To effectively manage the large number of products we sell, Walmart utilizes a hierarchical organizational structure to group products together at various levels of specificity. First, items are grouped into *Departments*, the broadest organizational level. Within each department, items are further subdivided into various *Categories*. Within each category, items are still further subdivided into smaller groups; however, departments and categories are the main groupings of interest in this paper.

In this paper, we focus on solving our Price Investment task with respect to the products we sell within a single department, our *Food & Consumables* department. We formalize the Price Investment problem below, as two sub-problems to be solved in order:

(1) Identify optimal allocation of our total budget across all product categories
(2) Within each category, generate optimal price recommendation for each individual component product

We developed and evaluated this system with respect to only one department as a proof-of-concept; however, our system can be easily applied to any of our other departments, which we leave as future work.

In this paper we refer to these sub-problems as phases 1 and 2 respectively. It was necessary to separate these aspects of Price Investment into two separate tasks in order to maintain computational tractability, given the large number of products in the Food & Consumables department. Consequently, budget allocation is done at the category level rather than the item level. The first phase identifies optimal category-wide investment budgets that are then used as constraints in the second price-recommendation phase within each category. We wish to solve both algorithmically in such a way that our portfolio and corresponding prices maximize both future sales and market share. The solutions can then provide reliable answers to questions such as:

- Should we invest more in products in category A or those in category B?
- What products should we focus on to improve category-wide sales and store traffic?
- What are the optimal prices for identified products that maximize demand volume?

1.2 System Architecture

The PRS is comprised of the following components (Figure 1):

- A weekly-refreshed *data loader*, which loads the most recently updated Walmart sales data into the system
- A *data pre-processor*, which performs standard data cleaning procedures
- A *model builder*, which forecasts the demand of given products for a given price markets
- A *portfolio optimizer*, which distributes our total available investment across our suite of product categories (2.2)
- The *price optimizer*, which produces optimal price recommendations for each individual product (2.3)
- *Post-assessment and Causal Inference Analysis*, to verify the efficacy of our pricing system (3.2)

2 METHODS

2.1 Data Preparation

Walmart groups its approx. 4700 US stores into 52 different Pricing Markets, which we also simply call *Markets* in this paper. These groupings are determined using customer cross-shopping patterns as well as store geographic locations. In keeping with both our EDLP strategy and the business practice outlines set by our founder, Sam Walton, our Central Pricing Strategy team sets the prices of items being sold in Walmart stores, using a variety of methods and systems like our Scenario Planner and Price Performance tools, among others. The price for any product sold by Walmart is set at the pricing market level to prevent discrepancies in price between similar stores. This standardization prevents Walmart stores from

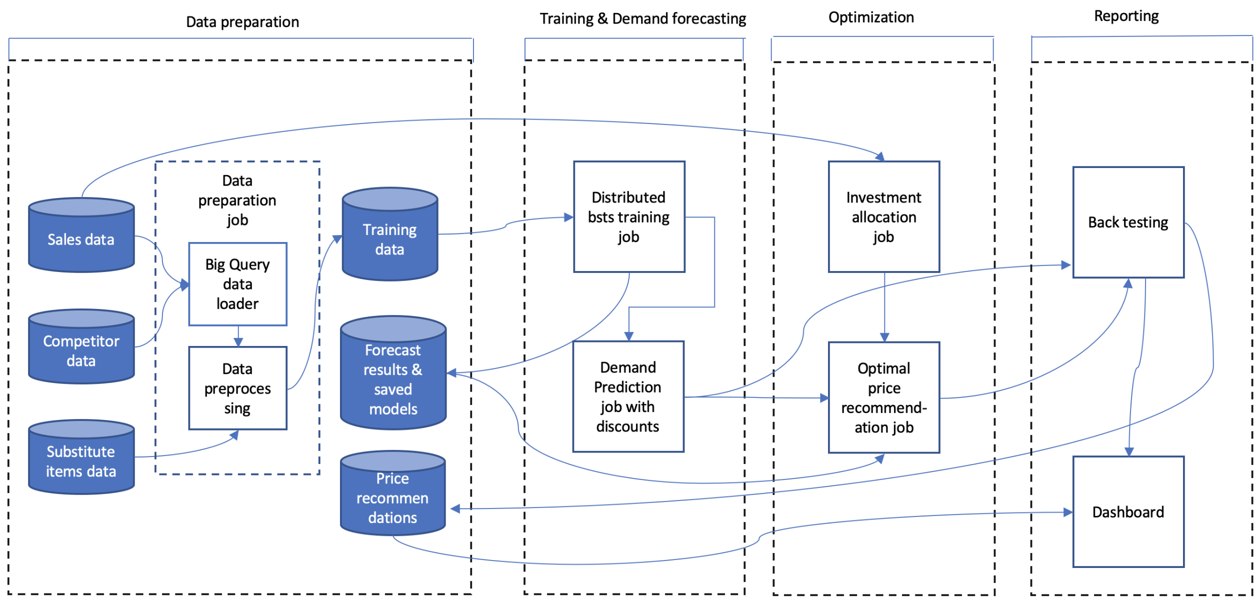

Figure 1: System Architecture

cannibalizing each other's business.

Walmart's mission and EDLP business strategy make it essential for the business to understand market conditions in order to make our own pricing decisions. To help inform our own pricing decisions, we use data describing the broader retail market and our competitors, sourced both from our internal research and from Nielsen [3]. This data, along with Walmart's internal point-of-sales information, allows us to produce optimal price recommendations for each item.

Because Walmart tracks such business data at a breadth of granularity, we first identified the correct data granularity to use for modeling. For example, an accurate forecasting model for predicting future item demand is an important part of the PRS (cf. 2.2.3-2.2.5), yet such a model can be formulated to observe data and make predictions at different levels of each of three different granularities:

- *Time*; e.g. "Do we want demand forecasts made for individual days, or at a weekly, monthly, or higher level?"
- *Product*; e.g. "Do we want demand forecasts made for each product individually, or for an entire product line or category?"
- *Geography*; e.g. "Do we want demand forecasts generated separately for each store, or do we want to forecast the aggregate demand across a whole price market, or even across the whole country?"

Our data on the retail sector and our competitors is at either the Item-Market-Week level, or the Item-Store-Week level, whereas Walmart internal data is available at the much finer single-transaction level. Walmart makes company-wide pricing decisions for each product at a quarterly timescale, and at the price-market level. Therefore, our finer-grain data was aggregated correspondingly for optimization (cf. 2.2.2, 2.3.2). However, demand forecasting is done for each product at a *weekly* timescale at the price-market level, rather than at the quarterly timescale, in order to extract predictive information from short-term dynamics such as calendar events (cf. 2.2.3-2.2.5), and then include it into our model. These weekly forecasts are then aggregated across items to the category level, and across weeks to the quarterly level before being fed to the optimizer.

2.2 Phase 1: Budget Allocation

Once the data has been appropriately cleaned and prepared according to the process outlined above, it is then used to solve the first of the two sub-problems outlined in (1.1): the budget allocation task across product categories. This task is naturally expressed as a portfolio optimization problem [4]. We assume that the recommended prices remain constant during each quarter (13-week); company-wide pricing decisions are made at this broad timescale because implementing increasingly dynamic pricing introduces higher operational costs in the brick-and-mortar context.

2.2.1 Definitions. Let C be the set of all product categories, and $\mathcal{P}$ be the set of price discount levels (i.e. percentage discounts) that can be applied to any category. In this step we consider applicable discounts only at the product category level, because the large number of individual products considered renders item-level optimization tasks computationally intractable. We assume that any discount level $i \in \mathcal{P}$ can be applied to any category $c \in C$

Let

$$d_i^c \in \{0,1\}, \forall i \in \mathcal{P}, \forall c \in C \tag{1}$$

where for some category c, d_i^c is an indicator variable representing whether or not a discount of level i is applied to c during the current quarter. We define,

$$\hat{V}_{i,c} := \text{Sample Avg. of } V_{i,c}\ \forall c \in C, \forall i \in \mathcal{P} \tag{2}$$

$$\hat{R}_{i,c} := \hat{V}_{i,c} W_c \left(1 - \frac{i}{100}\right) \tag{3}$$

$$\hat{R}^{\sigma}_{i,c} := \text{ Sample Var. of } V_{i,c} W_c^2 \left(1 - \frac{i}{100}\right)^2 \tag{4}$$

$$\hat{I}_{i,c} := \hat{V}_{i,c} O_c \tag{5}$$

as the set of estimators where $V_{i,c}$ is a distribution corresponding to the total demand volume (in units) for the following quarter, for category c and discount level i; W_c and O_c are average price and cost of the category before discount; $\hat{R}, \hat{I}$ and $\hat{R}^{\sigma}$ denotes the sample mean of revenue, investment and sample variance of revenue, respectively. We remark that the distribution of demand is a function of discount percentage (prices) and we have observed from the data that the variance of the distribution increases as we deviate from the mean of prices seen in the model data.

Let $\tilde{R} \in \mathbb{R}$ be the revenue target we wish to meet for the following quarter, which is chosen based on market growth, business expectations, and year-over-year sales. This target is set by Walmart's business teams. Let $B \in \mathbb{R}$ be the total budget the business can spend on operational costs across all product categories, during the following quarter. This is also a fixed value provided by our business teams.

2.2.2 *Formulation.*

With these, we wish to perform the following constrained optimization:

$$\begin{aligned}
\min \quad & \sum_{c\in C} \sum_{i\in\mathcal{P}} \hat{R}^{\sigma}_{i,c} d_i^c && \text{(6a)}\\
\text{s.t} \quad & \sum_{c\in C} \sum_{i\in P} \hat{I}_{i,c} d_i^c \leq B && \text{(6b)}\\
& \sum_{c\in C} \sum_{i\in P} \hat{R}_{i,c} d_c^i \geq \tilde{R} && \text{(6c)}\\
& \sum_{i\in\mathcal{P}} d_i^c = 1 \qquad \forall c \in C && \text{(6d)}\\
& d_i^c \in \{0,1\} \qquad \forall i, c && \text{(6e)}
\end{aligned}$$

Essentially we wish to find a set of discount levels, exactly one for each category, that minimizes the volatility of the following quarter's total revenue yielded from applying each discount level to its respective category, in such a way that our total expected investment is at most the same as our total allocated budget, and our expected total revenue across all categories at least meets our revenue target. Note that in this phase, applying a *discount* to a category amounts to increasing our budget investment into this category. Minimizing revenue volatility allows us to meet these targets while also managing our investment risk.

Solving the above optimization problem requires knowledge of the distributions of $R_{i,c}$ and $I_{i,c}$, both of which depend on the distribution of $V_{i,c}$. Thus, it is necessary to determine both the mean and variance of the distribution $V_{i,c}$. To this end, we make use of the sample distribution of the demand from the BSTS to empirically estimate the mean and variance of $V_{i,c}$ at different discount values.

2.2.3 *Demand Forecasting.*

To obtain this information, we model the relationship between price changes and future demand volume. However, the demand of a particular item in a given price market depends not solely on its price, but also on other factors such as competitor prices, the difference between Walmart's prices and those of the overall market, sales, number of units sold in rest of the market, etc. Additionally, the time-indexed nature of our task and data meant that calendar and seasonal data such as week-of-the-year, month-of-the-year, quarter-of-the-year, and year-over-year (YoY) seasonality features also play an important role in influencing item demand, and thus were necessary to include in the modeling procedure.

Thus, in order to understand how demand volume evolves over time with respect to price changes, it was necessary to model this entire system via a demand forecasting model.

2.2.4 *Uncertainty Quantification via Bayesian Inference .*

The formulation in (2.2.1) however does not make use of simple point-forecasts, but rather is dependent on the *posterior predictive distribution* of the demand forecasts, conditioned on the training data; this distribution in turn determines the distributions of $R_{i,c}$ and $I_{i,c}$. *Bayesian Inference* is a well-studied and developed framework for statistical inference that allows practitioners to both intuitively incorporate prior beliefs about certain data into the modeling process, and also obtain comprehensive uncertainty estimates about predictions. Bayesian approaches provide such uncertainty quantification by directly producing posterior predictive distributions, rather than the point forecasts produced by traditional *Frequentist* methods. Bayesian methods can also be used to easily produce point forecasts by simply taking the mean of the posterior predictive distribution output. Because these tasks are naturally incorporated into Bayesian analysis and are thus easier and more intuitive to perform, Bayesian methods are often preferred for use in contexts requiring uncertainty estimation [5].

2.2.5 *Bayesian Structured Time Series .*

To generate the demand-volume forecasts as well as uncertainty estimates, we use Bayesian Structural Time Series (BSTS), a Bayesian time-series model often used for feature selection, time series forecasting, and causal impact inference. Unlike traditional time-series models like ARIMA, BSTS naturally allows us to quantify posterior uncertainty by providing a complete posterior predictive distribution as output, rather than point forecasts. [6]

BSTS is an example of a *structural* or *state space* model. Such models are defined by two equations. The first, called the *observation equation*,

$$Y_t = Z_t^T \alpha_t + \epsilon_t \tag{7}$$

describes the relationship between our observed target, Y_t, and vector of latent variables α_t, called the *latent state* of the system. The second equation, called the *transition equation*, models the evolution of this latent state over time:

$$\alpha_{t+1} = T_t \alpha_t + R_t \eta_t \tag{8}$$

In the above formulation, the vectors Z_t, T_t, R_t are *structural parameters* that are constructed manually according to the evolution dynamics of the modeled system. These parameters render structural models extremely flexible; many classical time-series models such as autoregressive models can be expressed equivalently as structural models. This flexibility makes structural modeling an extremely useful approach.

We use the following structural time-series model incorporating three state components: a *local-level trend* μ_t, a seasonal pattern τ_t [7, 8], and external regressors $\mathbf{x}_t$:

$$Y_t = \mu_t + \tau_t + \beta^\top \mathbf{x_t} + \epsilon_t, \epsilon_t \sim \mathcal{N}(0, \sigma_\epsilon^2) \tag{9a}$$

$$\mu_{t+1} = \mu_t + \eta_t, \eta_t \sim \mathcal{N}(0, \sigma_\eta^2) \tag{9b}$$

$$\tau_{j,t+1} = \tau_{j,t} \times \cos(\lambda_j) - \tau_{j,t}^* \times \sin(\lambda_j) + \omega_{j,t} \tag{9c}$$

$$\tau_{j,t+1}^* = \tau_{j,t}^* \times \cos(\lambda_j) - \tau_{j,t} \times \sin(\lambda_j) + \omega_{j,t}^* \tag{9d}$$

$$\tau_t = \sum_{j=1}^{k} \tau_{j,t} \tag{9e}$$

$$\lambda_j = 2\pi j / s \tag{9f}$$

where $j = 1, .., k$ is the j-th seasonal frequency, s is the length of the longest seasonal cycle in number of timesteps, and Y_t is our observed target. The local-level trend μ_t models the evolution of the latent state of the system, which as described in Eq. (9b), is assumed to evolve following a random walk in levels. This choice reflects a belief in no strong upward or downward trend in demand volume at a short-term weekly level that is not related to seasonality or changes in the values of the features included. Although the large number of category-price market combinations made this assumption difficult to exhaustively check in each case, we believe it is appropriate for our task given our domain knowledge. We also impose a *Spike-and-Slab* prior on our regression coefficients, which enables automatic feature selection via parameter shrinkage [9]. We impose an inclusion probability of 1 on the price feature and 0.5 on all others. Additionally we set the elements of the prior mean vector to ±0.5, with the sign of each element determined by the assumed directionality of the relationship between the corresponding feature and demand volume.

As discussed in (2.1), modeling was done at the item-week level within a single price market. In our model, for week t, the observed target is the weekly demand volume for a given UPC, and the feature vector x_t includes the UPC's current price, the weekly average price of the product in competitor stores, the weekly difference between Walmart's price for the product and the average competitor price, the weekly total number of units sold of the product across the entire market, the cost of the product for Walmart, the percent level of any promotions applied to the product by Walmart this week, and a set of indicator variables for various special calendar events (such as holidays), each representing whether or not the event occurred during week t. Our model also incorporates weekly, semi-annual, and quarterly seasonality. As is typical with Bayesian statistical models, our forecasting model is fit using a Markov Chain Monte Carlo (MCMC) method.[6]

Because demand modeling is done at the UPC level, we assume *causal independence* between demand volume for all categories and items when we produce the mean and variance estimates for the demand volume for a quarter. We aggregate results across individual items and weeks to the category-quarterly level for use in the optimization of Phase 1. Our demand forecasting model was implemented using the bsts R package [10]. Although producing forecasts for individual products is computationally intensive, we note that forecasting for individual products can be run in parallel.

Performing the optimization in (2.3) using estimates from the above model, we arrive at an optimal set $\mathcal{D} = \{\tilde{i}_c\}_{\forall c \in C}$, where $\tilde{i}_c$ is the single discount level for category c to be applied. Given $\mathcal{D}$, we can now compute for each category c, the expected total investment allocation $\tilde{I}_c := \sum [\hat{I}_{\tilde{i},c,1}]$ for that category.

2.3 Phase 2: Pricing

The next phase involves, within each category, generating optimal price recommendations for each UPC within the category. We use the category-wide investment totals $\tilde{I}_c$ from the previous problem and formulate Phase 2 as another constrained optimization problem.

2.3.1 Definitions. Let $\mathcal{M}_c$ be the set of UPCs within category c, and $\mathcal{P}_m$ be the set of possible price points for UPC $m \in \mathcal{M}_c$.

Let $\hat{R}_m(p_i), \hat{I}_m(p_i), \hat{V}_m(p_i)$ be estimators respectively for the next quarter's revenue, investment, and demand volume for a product m at price point p_i. Let $\tilde{V}_c$ be the target demand volume for the following quarter.

2.3.2 Formulation.

$$\max \quad \sum_{m \in \mathcal{M}_c} \sum_{p_i^m \in \mathcal{P}_m} \hat{R}(p_i^m) \tag{10a}$$

$$\text{s.t} \quad \text{Card } \{p_i^m\}_{i \in \mathcal{P}_m} = 1 \qquad \forall m \in \mathcal{M}_c \tag{10b}$$

$$\sum_{p_i^m \in \mathcal{P}_m} \hat{I}(p_i^m) \leq \tilde{I}_c \tag{10c}$$

$$\sum_{m \in \mathcal{M}_c} \sum_{p_i^m \in \mathcal{P}_m} \hat{V}(p_i^m) \geq \tilde{V}_c \tag{10d}$$

$$p_i^m \in \mathcal{P}_m \qquad \forall m \in \mathcal{M}_c \tag{10e}$$

Thus in this optimization we attempt to maximize our total revenue for the category. The above problem is solved for each category independently with category-wide investment constraints $\tilde{I}_c$ given by the results of the allocation task in Phase 1.

2.3.3 Product Substitution. *Product Substitution*, or simply *substitution*, is the phenomenon in which two different products are seen as equivalent or similar by customers, thereby implying a degree of dependence on consumer shopping patterns with respect to one item, on shopping patterns with respect to the other. Therefore, substitution is an important consideration for a pricing system implemented in the retail context, as lowering the price of one item may simply increase its demand volume at the cost of that of another similar item in a store's inventory, leading to *cannibalization*.

Instead of accounting for product substitution explicitly in our pricing model, we consider in our formulation Walmart's internal business rules for product categorization and pricing, which require that products belonging to the same *line*[1] have the same price, and that higher *ladder*[2]product variants have a lower price-per-UOM (Unit of Measure) than lower ladder variants. These rules ensure standardization of pricing across similar items, preventing cannibalization, thus removing the need to explicitly account for substitution in our optimization. These pricing rules, as well as other business constraints on cost, margin , inventory, etc., are considered in our formulation to produce feasible price recommendations. In rare cases, data anomalies rendered these constraints impossible to simultaneously satisfy (consider the case where a 4-pack item is already priced lower than its 2-pack variant; we cannot as a rule raise the price of the 4-pack, but lowering the price of the 2-pack may result in a suboptimal solution). In such cases we omit some of these constraints to solve the optimization, and log them as anomalous for later analysis.

3 SYSTEM PERFORMANCE EVALUATION

3.1 Demand Forecasting Model Evaluation

The performance of our demand-forecasting model is first evaluated through a backtest using past data, with respect to its accuracy both when prices remain stable, and when price changes are applied. To see whether and to what degree the model gives an accurate projection of Walmart stores' demand volume against actual demand volume, backtesting is conducted using data collected from two prior periods during which price adjustments were made, which we call *investment waves*. These backtests were only conducted on a small number of categories, and were intended only as an initial validation of our forecasting model's performance before incorporation into the larger PRS. More robust evaluation of our forecasting model was conducted as a part of our system-wide experimental evaluation (3.2).

Our backtest was intended to answer the following questions about our forecasting model:

- Is there a predicted unit-lift (demand volume increase) after a price change for each category? How much is the unit-lift?
- Is the forecast different from actual performance? If so, by how much?

The result of this backtest is shown in Table 1, in which the top five performing categories for each wave are chosen for display. In this test, we compared the total actual demand volume lift (over the previous year) with the total predicted volume lift for all of invested items in each category over the previous year. We use *Mean Absolute Percentage Error*, or *MAPE* as our measure of model accuracy [11]. As shown in Table 1, our model generally exhibited high prediction accuracy, with MAPEs not exceeding 3%, and corroborates our expectation that price reductions result in demand volume lift. Although we noticed some categories exhibited negative unit lift after price change, digging deeper into the data we found that this was caused by a general decline in demand for these categories.

3.2 Live Price Investment Experiments

Live experiments were necessary to examine the efficiency of the price changes determined by our system, and are used to examine the performance of all components of our system in a realistic setting. In November 2019, during a single investment wave, we implemented price recommendation for one quarter, for two separate price markets, producing optimal investment allocations and prices for potential items, which were applied to 500 individual stores.

To evaluate the live price investment performance, we focused on measuring the MAPE of our demand volume forecasts and the effects of price changes on demand volume. We have built a dashboard and generated weekly reports on the following key metrics: actual unit year-over-year lift (%), forecast unit year-over-year lift (%), MAPE, actual vs. forecast year-over-year unit lift (%) etc. To analyze the effectiveness of price changes, we used the causal inference functionality provided by the bsts R package used to develop our volume forecast model, instead of using randomized experiments (i.e. A/B tests). Because of the high customer sensitivity to item prices, we don't make use of A/B testing to avoid introducing price discrepancies among stores in the same pricing market, and thus negatively impacting customer experiences and confounding our analysis of the price-demand relationship. Additionally, A/B tests are costly to implement in physical stores due to labor costs. Therefore, to understand the effectiveness of price change events on demand volume, we predict the counterfactual by comparing the observed data during the pre-event and the post-event periods. We also identified similar price markets as *control price markets* for each of the two markets we tested, to avoid relying solely on pre-event data to validate the price change impact.

The results of these experiments are displayed in Table 2, containing the top 5 performing categories with respect to post-treatment demand-lift compared to the control, as well as aggregated results across all categories in the Food & Consumables department. As is shown, for both price markets, our aggregate post-treatment Actual Unit YoY Lift in demand volume is greater than the control, exhibiting a 4.15% and 19.25% improvements over control respectively for each of the two price markets tested.

We now discuss the detailed performance and casual impact analyses for the two selected categories:

Analysis I:

- As shown from Figure 2, during the post intervention period, the demand had an average value of approx. 1.05B. By contrast, in the absence of an intervention, we would have expected an average demand of 0.73B. The 95% confidence interval of this counterfactual prediction is $[0.67B, 0.79B]$.

[1]A product `line` in Walmart is a grouping of similar items (e.g. plain and vanilla yogurt both belong to the same line)
[2]Same items with bigger pack (e.g. 4-pack vs. 6-pack of Heinz Ketchup)

	Category (anonymized)	Actual pre. vs post (%) [1]	Model pre. vs post(%) [2]	MAPE (%)[3]	Invested Amount pct (%)[4]	Avg Discount(%) [5]
WAVE 7B	Category A	33.23	33.05	0.42 ± 0.81	4.82	10.91
	Category B	13.35	14.21	0.71 ± 1.14	2.18	10.85
	Category C	−3.28	−4.12	0.76 ± 0.69	0.69	9.57
	Category D	28.87	27.96	1.32 ± 0.21	7.84	9.15
	Category E	−5.63	−7.02	2.69 ± 0.42	2.81	11.45
WAVE 11	Category C	−1.58	−2.17	1.01 ± 1.24	6.69	10.13
	Category F	21.67	22.43	0.28 ± 0.71	3.19	15.00
	Category J	30.26	21.34	0.43 ± 1.25	10.51	9.50
	Category D	−1.21	−4.13	0.52 ± 0.91	0.93	5.99
	Category B	−3.97	−4.05	0.74 ± 0.63	2.52	11.96

[1] *Actual pre. vs. post:* 13-week actual demand of given category before price change vs. after price change
[2] *Model pre. vs. post:* 13-week actual demand of given category before price change vs. forecast demand after price change
[3] *MAPE:* Mean absolute percentage error of demand model
[4] *Invested Amount (%):* Invested dollar-amount percentage for a given category
[5] *Avg Discount (%):* Average discount of all invested items for a given category

Table 1: Model Performance on Wave 7B and Wave 11

	Category (anonymized)[1]	Avg Discount (%)[2]	MAPE (%)[3]	Actual Unit YoY Lift (%)[4]	Forecast Unit YoY Lift (%)[5]	Control Unit YoY Lift (%)[6]
WAVE 15, Price Market 1	Category A	14.25	1.87	18.74	17.89	−1.09
	Category B	24.32	1.01	31.36	33.33	22.90
	Category C	8.17	7.68	26.76	17.25	11.25
	Category D	18.42	2.78	11.32	14.33	−9.27
	Category G	27.23	6.26	21.37	29.28	14.09
	Average	17.68	7.22	13.09	5.63	8.94
WAVE 15, Price Market 2	Category A	19.17	1.65	14.38	15.24	9.35
	Category B	20.32	3.21	18.35	22.26	11.03
	Category C	21.18	4.67	24.34	20.23	7.35
	Category H	24.67	3.27	4.25	1.89	−6.07
	Category I	22.27	3.07	41.29	38.57	−10.76
	Average	19.26	8.37	21.82	11.03	2.57

[1] *Category (anonymized):* Anonymized category name
[2] *Avg Discount* (%)*:* Average discount of all invested items for a given category
[3] *MAPE:* Mean Absolute Percentage Error of demand volume forecasts
[4] *Actual Unit YoY Lift* (%)*:* Average actual volume lift over previous year actual volume for a given category
[5] *Forecast Unit YoY Lift* (%)*:* Average forecast volume lift over previous year actual volume for a given category
[6] *Control Unit YoY Lift* (%)*:* Average actual volume lift over previous year actual volume for a given category in identified control market

Table 2: Model Performance for Wave 15, for two Price Markets

Subtracting this prediction from the observed demand yields an estimate of the causal effect the intervention had on the demand. This effect is $0.32B$ with a 95% interval of $[0.26B, 0.38B]$.

- Summing up the individual data points during the post-intervention period, the demand had an overall value of $13.64B$. By contrast, had the intervention not taken place, we would have expected a sum of $9.46B$. The 95% interval of this prediction is $[8.67B, 10.28B]$.
  The above results are given in terms of absolute numbers. In relative terms, the demand showed an increase of +44%. The 95% interval of this percentage is [+35%, +53%].
  This means that the positive effect observed during the intervention period is statistically significant and unlikely to be due to random fluctuations. It should be noted, however, that the question of whether this increase also bears substantive significance can only be answered by comparing the absolute effect ($0.32B$) to the original goal of the underlying intervention.
  The probability of obtaining this effect by chance is very small (Bayesian one-sided tail-area probability $p = 0.001$). This means the causal effect of price can be considered statistically significant.

Analysis II:

- As shown from Figure 3, during the post-intervention period, the demand had an average value of approx. 2.96M. By contrast, in the absence of an intervention, we would have expected an average demand of $1.78M$. The 95% interval of this counterfactual prediction is $[1.56M, 2.01M]$. Subtracting this prediction from the observed demand yields an estimate

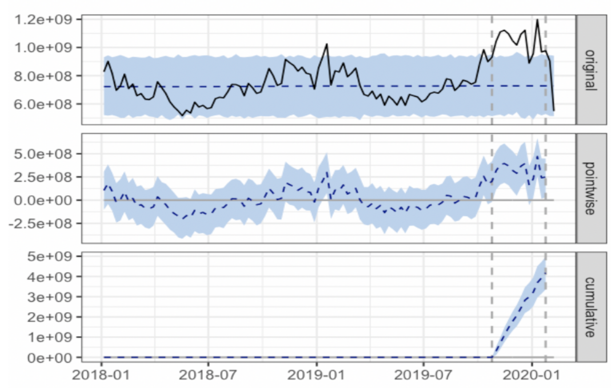

Figure 2: Price Impact Visualization on Category: Mac and Cheese in Dallas Market

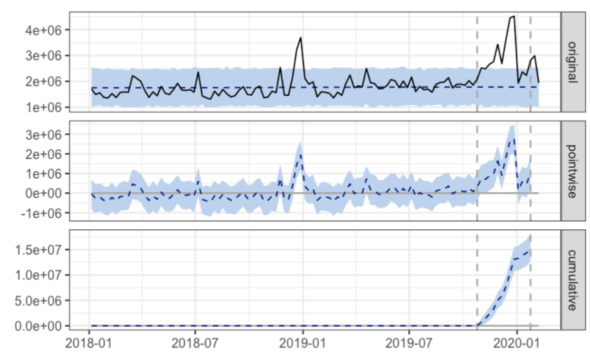

Figure 3: Price Impact Visualization on Category: Mustard Ketchup in Kansas City Market

of the causal effect the intervention had on the demand. This effect is 1.18M with a 95% interval of $[0.95M, 1.40M]$.

- Summing up the individual data points during the post-intervention period, the demand had an overall value of $38.48M$. By contrast, had the intervention not taken place, we would have expected a sum of $23.15M$. The 95% interval of this prediction is $[20.23M, 26.13M]$.
  The above results are given in terms of absolute numbers. In relative terms, the demand showed an increase of +66%. The 95% interval of this percentage is [+53%, +79%].
  This means that the positive effect observed during the intervention period is statistically significant and unlikely to be due to random fluctuations. It should be noted, however, that the question of whether this increase also bears substantive significance can only be answered by comparing the absolute effect ($1.18M$) to the original goal of the underlying intervention.
  The probability of obtaining this effect by chance is very small (Bayesian one-sided tail-area probability ($p = 0.001$). This means the causal effect of price can be considered statistically significant.

3.3 System Scalability

Our system is designed such that results for multiple categories of products across different price markets can be computed independently, thereby enabling parallelism to ensure faster delivery of results. All the results produced by our system are stored in GCP (Google Cloud Platform) [12] to provide high availability for downstream systems. The final recommended price of each item in each category and department is provided to business and implemented in stores through our internal product system.

The pipeline for executing price recommendation is built on our internal *TrainMe* platform. This platform is built over GCP to run machine learning models in a distributed environment. It ensures high availability of compute resources and manages these resources effectively. The main aspects of our highly scalable workflow are:

- Distribution on VMs (Virtual Machines): Each category for a pricing market runs in one VM independent of other categories. We achieve parallelization by running the entire department at once by using multiple VMs.
- Parallelization within a VM: Within a category in a VM, we use multi-processing to achieve parallel computation for each UPC model. Each VM has 32 VCPUs which means we build 32 UPC level models in parallel.

These two strategies have helped in bringing down the compute time for price recommendation and has been proved to be cost effective. Data is distributed and read per category. Result and models are saved back to the GCP bucket, ensuring high availability of our results for downstream systems. The entire process is automated and requires minimum supervision. The Train Me platform dynamically provisions VMs and shuts them after use, providing cost effectiveness.This is a highly scalable architecture and has helped in bringing down the compute time and thereby the costs incurred. Our pipeline is capable of producing results for 57 categories of products in 52 mins at the cost of $68.

4 CONCLUSION

We introduce a novel price recommendation system for items in our Food & Consumables department. This system, which we call the PRS is comprised of a demand forecasting model, two optimization steps, and causal inference analysis for validation. We validated our demand forecasting model with multiple backtests to measure the demand volume prediction accuracy with respect to change in price. To evaluate system performance as a whole, we performed two live experiments in which we made price adjustments using our new system. The results from these experiments demonstrate the efficacy of our method over the currently implemented rule-based pricing system used by Walmart. We have also implemented the system at Walmart's large scale using distributed computing. To our knowledge this is the first actual live experimental results of the prescriptive price optimization in a brick-and-mortar retail setting. As future work, we wish to explore applying our system to other departments and deploy our PRS across all stores and product departments within Walmart. We believe systems like ours which can learn to make decisions in an automated manner will play a major role as we take a step closer to the digital transformation of retail stores. Apart the benefits in terms of revenue, this work generated systematic price variations which are essential to understand the relationship between price elasticity and substitution. With a better understanding of the price-demand relationship, we at Walmart will be able to serve the customer better with optimally priced high quality products.

REFERENCES

[1] Kimberly Lewis. The advantage brick-and-mortar stores will always have over online retail, Sep 2018. https://www.forbes.com/sites/forbesnonprofitcouncil/2018/09/04/the-advantage-brick-and-mortar-stores-will-always-have-over-online-retail/#8b8b750772db.

[2] Hanssens Srinivasan, Pauwels and Dekimpe. Who benefits from price promotions?, Aug 2014. https://hbr.org/2002/09/who-benefits-from-price-promotions.

[3] About nielsen global connect and nielsen global media. *What People Watch, Listen To and Buy.* https://www.nielsen.com/eu/en/about-us/.

[4] Harry Markowitz. Portfolio selection. *The journal of finance*, 7(1):77–91, 1952.

[5] G.B. King, A.E. Lovell, L. Neufcourt, and F.M. Nunes. Direct comparison between bayesian and frequentist uncertainty quantification for nuclear reactions. *Physical Review Letters*, 122(23), Jun 2019. http://dx.doi.org/10.1103/PhysRevLett.122.232502.

[6] Kay H. Brodersen, Fabian Gallusser, Jim Koehler, Nicolas Remy, and Steven L. Scott. Inferring causal impact using bayesian structural time-series models. *Annals of Applied Statistics*, 9:247–274, 2015.

[7] Steven L. Scott. add.trig: Trigonometric seasonal state component in bsts: Bayesian structural time series, Sep 2019.

[8] Andrew Harvey. Forecasting with Unobserved Components Time Series Models. In G. Elliott, C. Granger, and A. Timmermann, editors, *Handbook of Economic Forecasting*, volume 1 of *Handbook of Economic Forecasting*, chapter 7, pages 327–412. Elsevier, 2006.

[9] Hemant Ishwaran and J. Sunil Rao. Spike and slab variable selection: Frequentist and bayesian strategies. *Ann. Statist.*, 33(2):730–773, 04 2005. https://doi.org/10.1214/009053604000001147.

[10] Package bsts. https://cran.r-project.org/web/packages/bsts/index.html.

[11] P. M. Swamidass, editor. *MAPE (mean absolute percentage error)MEAN ABSOLUTE PERCENTAGE ERROR (MAPE)*, pages 462–462. Springer US, Boston, MA, 2000. https://doi.org/10.1007/1-4020-0612-8_580.

[12]