SMU Data Science Review

Volume 3
Number 3 *Fall 2020*

Article 4

2020

Price Optimization for Revenue Maximization at Scale

Nikhil Gupta
Southern Methodist University, mywork.ng@gmail.com

Massimiliano Moro
Southern Methodist University, massimiliano.moro@gmail.com

Kailey A. Ayala
Southern Methodist University, kayala@mail.smu.edu

Bivin Sadler
Southern Methodist University, bsadler@smu.edu

Price Optimization for Revenue Maximization at Scale

Nikhil Gupta, Massimiliano Moro, Kay Ayala, Dr. Bivin Sadler

Master of Science in Data Science, Southern Methodist University,
Dallas, TX 75275 USA
{guptan, mmoro, kayala, bsadler}@mail.smu.edu

Abstract. This study presents a novel approach to price optimization in order to maximize revenue for the distribution market of non-perishable products. Data analysis techniques such as association mining, statistical modeling, machine learning, and an automated machine learning platform are used to forecast the demand for products considering the impact of pricing. The techniques used allow for accurate modeling of the customer's buying patterns including cross effects such as cannibalization and the halo effect. This study uses data from 2013 to 2019 for Super Premium Whiskey from a large distributor of alcoholic beverages. The expected demand and the ideal pricing strategy to maximize revenue for the business is analyzed. This study shows that models with cross effects can improve the forecasting capability by 39% over naïve models and by 14% over models without cross effects. Using these more accurate models, an optimal price point was calculated, which forecasted an increase in revenue by 30%. While the techniques presented in this paper have been validated for the distribution market of alcoholic beverages, they don't rely on any domain specific knowledge from this industry, and thus can be applied to other distribution markets for non-perishable products.

1. Introduction

In an ever-competitive world, companies need to find optimal pricing for their products in order to maximize business objectives such as revenue and profits. Revenue for a product depends on a concept called "price elasticity" [1], [2]. When price is high, demand drops, but more revenue is made per sale. When price is low the opposite effect is observed. Hence, a business must find the ideal price point for their products in order to maximize revenue. This concept becomes much more complex when cross effects such as cannibalization and halo effects are taken into consideration. Cannibalization indicates that the price of a product depends not just on the price of that product but also similar products. A drop in the price of a product can cause switching behavior in customers leading to reduction in the revenue of other products. The halo effect refers to the customer preference for a brand due to a positive experience with another product from the same brand [3]. This implies that a company could charge a premium for certain products due to positive image from other products of the same brand.

There are several factors that further complicate the process of determining the optimal pricing strategy. For example, cross effects are not only non-linear but asymmetric as well. How product "A" influences product "B" is not the same as how product "B" influence product "A" [4]. Moreover, the change in the demand of a product may not only be influenced by switching behavior, but instead caused by other factors such as stockpiling (customer accelerates purchases at a discounted pricing) or a true increase or decrease in demand [3]. This can confound the results of evaluating the long-term impact of a pricing strategy. Furthermore, evaluating cross effects across hundreds of products becomes a herculean task.

Recent advances in data mining and forecasting techniques provide a glimmer of hope. Association mining techniques (such as those used by Netflix and Amazon to make recommendations) can be used to evaluate the most influential set of products that can have an impact on the revenue of other products. Multivariate time series analysis, and deep neural networks (such as Recurrent Neural Networks - RNN and Long Short-Term Memory - LSTM) can consider a complex set of interdependent and sequential variables to unveil patterns as well as provide a more accurate prediction of demand. The advent of Automated Machine Learning (AutoML) can help model the demand for products at scale. Additionally, advances in optimization methods such as Bayesian optimization plus the increase in computing power can help navigate different pricing scenarios more efficiently to find the ideal price point for various products jointly.

Using these techniques, this study provides an answer to one of the most important question for a company – what is the ideal pricing strategy for all products to drive up demand and maximize revenue? While this research focuses on the distribution market for alcoholic beverages, the techniques applied in this study are not industry specific and are generic enough to be applied to other distribution industries.

2. Literature Review

2.1 Elasticity and Revenue Management

Revenue management is defined as the science of maximizing revenue by controlling the price of the product [5]. This concept is based on the theory of price elasticity, [1] which states that as the price of a product is reduced, its demand will increase. Initially the increase in demand will be enough to offset the decrease in price resulting in an increase in revenue. However, after a certain price point, the increase in demand will not be able to make up for the decrease in price and the revenue will eventually start to fall. This points to the existence of an optimal price point for every product to maximize revenue [2].

In terms of increasing revenue, price cuts are the most effective medium, even more so than advertisement [3]. This is an especially important topic for managers since they want to know whether a drop in price is bringing in new customers or simply providing discounts to customers who would have paid the non-discounted price anyway [3]. On the other end of the spectrum, increasing price to drive revenue can still work since core customers are likely to continue purchasing products when price is increased. This can offset the loss of revenue from fringe customers who might stop buying if price increases too much [4].

2.2 Cannibalization and Cross-Elasticity

Any sales bump derived from price reduction can be temporary (customer stockpiling at the reduced price leading to a reduction in future sales), lasting (true increase in demand due to the reduced pricing) or the result of shifting revenue from one product to another (switching behavior, e.g. cannibalization) [3]. It is important to decipher between the various sources of increase in demand. This is because some of them are temporary and only lead to an acceleration of revenue and not a true increase [3].

Cannibalization is of key interest in this study. This has been well studied in various industries such as the airline and hotel industry [3], [5]. Cannibalization is defined as the reduction of sales in one product due to another product [6]. This can be traced to the "cross-elasticity of demand theory" which essentially states that the change in demand in one product is influenced by the change in price of another product [7]. However, the concept of asymmetric price elasticities can lead to more complexity in determining cross-elasticity [4].

The methods proposed in this research are data driven. Hence, they capture the customer's buying pattern and the underlying reason for any potential increase in demand due to discounts. Any inherent stockpiling or purchase acceleration behavior will be captured in the model as a drop in demand following the stockpiling event. Switching behavior is studied by adding the pricing of the most influential products in the models. This study leverages association mining literature to determine the most influential products that can impact the demand for another product. True increase in demand will be visible if none of the above effects are found.

2.3 Modeling Cannibalization

Not much literature has been devoted to modeling revenue management taking price elasticity into consideration [3], [5]. Most studies focus on assuming a linear dependence of demand on price [8]. In the best case, saturation effects at extremely low and high price points are taken into consideration using a "probit" approximation [5]. However, the nature of cross-effects may be non-linear in nature and not much research seems to have been done in this regard.

Previous literature has focused predominantly on perishable products (a product whose utility expires after a certain time) such as the airline industry and hotel industry. In these products, there is a strict deadline by which the product needs to be sold and after this deadline is crossed, the utility of the product diminishes to zero. This means that as the deadline approaches, companies may need to discount their products heavily to make sure that they are sold. In contrast, this study is focused on the non-perishable product space where the market dynamics may be very different. At best, heavy discounting may not be needed in order to sell the product and at worst, the product may consume unnecessary warehouse space.

Moreover, some of the literature is based on an expert system approach wherein the opinion of experts is used to construct appropriate pricing curves for elasticity [5]. This approach lacks scalability. Additionally, not every company has access to experts, and even if they do, the intuition may be limited to capturing linear elasticity patterns as explained in economic theory [5]. This may in turn miss out on the subtle non-linearities involved in the pricing curves.

Additionally, most of the literature is concentrated on the direct consumer market where the solution in one industry such as the airline industry may not be applicable to the hotel industry [5], [9]. This study is focused on the distributor market. When switching from a direct consumer market to a distribution market, this effect can be exacerbated. For example, when a company sells directly to a consumer, brand switching by the consumer is a problem since it is a loss of revenue. However, when a distributor sells products from multiple brands, this is not an issue since they still get the revenue. In addition, switching stores in the consumer market is much easier than in the distribution market due to the volume of choices available.

While private research may have been done in the area of the impact of price on demand in the distribution market, scholarly literature on this topic is scant. The goal of this paper is to shed light into the nuances of price elasticity and cross effects as it pertains to the distribution market.

2.4 Product and Customer Segmentation

This study focusses on creating an optimal pricing strategy for the distribution market of alcoholic products. Previous studies in this area concluded that demand needs to be modeled on a per product basis due to different buying patterns, product preferences and tolerance to price changes [10], [11]. This has been backed up in previous studies that noted the change in sales due to change in pricing can vary across brands and products [12], [13], [14]. In a consumer market that is focused on thousands of customers with sporadic purchasing patterns, modeling on a per customer basis is generally not possible. Instead, aggregate effects are considered by combining the revenue of products across all customers. This can lead to a loss of information such as the buying patterns and price sensitivities of individual customers. However, a distribution market is inherently different. The number of customers is limited, and the sales are more frequent and in larger volumes. Hence modeling on a per customer basis is feasible and gives a nuanced model which captures the buying patterns of individual customers. This is important to consider since some customers may be more price sensitive than others.

This paper proposes a method to compute the price point at which a product should be offered to individual customer of the distribution company. By offering different price points for the same products to different customers based on their purchase propensity, a distribution company will be able to maximize their revenue.

2.5 Demand Models

On the one hand, it has been demonstrated that a Vector Autoregressive (VAR) based demand forecasting framework produced good results [3]. On the other hand, "no one sized fit all" model exists for the distribution of alcoholic beverage market [10], [11]. The type of model that worked for one customer-product combination did not work well for another. Hence, a variety of modeling techniques were tried such as Naïve models, statistical models (e.g. ARMA, ARIMA, Seasonal ARIMA, VAR, Signal Plus Noise, Multiple Linear Regression with Correlated Errors), Deep Learning models (based on variants of Long Short Term Memory) and "Ensemble" techniques [10], [11]. These authors also noted that due to limited amount of available demand data (84 observations in their cases – 7 years sampled on a monthly basis) the deep learning approaches tend to overfit. In addition, while both these studies focused on less than five customers and products, they indicated that the ability to scale would need an automated machine learning system (AutoML) [10], [11].

Another popular forecasting technique that has gained popularity recently has been made available through the Prophet library from Facebook [15]. This library provides a way to fit non-linear trends with multiple seasonality and has been shown to be robust to missing data and data containing outliers which would be typically found in real world scenarios.

2.6 Automated Machine Learning (AutoML) for Time Series

Due to recent advances in computing, and data science, as well as the democratization of machine learning, several AutoML systems have been introduced in the marketplace. One such framework is Auto-sklearn [16] which is built on the popular scikit-learn framework in python. Demand models are inherently time series based as the demand is a function of time, in addition to other exogenous variables. Although the scikit-learn framework has support for time series-based sliding window resampling methods as well as imputing and feature engineering methods, they do not support the traditional autocorrelation-based model (such as ARIMA and Vector Auto Regressive models) [17].

Similarly, a recent entry into the world of AutoML is PyCaret, but it also does not support time series models [18]. Other systems such as the H2O.ai Driverless AI support time series forecasting, but these are paid products costing thousands of dollars [19].

Hence, in order to truly scale this across several customers and products, this study develops an AutoML framework that can support a variety of machine learning models and methods such as Seasonal ARIMA, Vector Auto Regressive models, XGBoost, and Stacking. Fortunately, libraries such as Statsmodels have extensive support for time series analysis [20]. Based on research, using available libraries and frameworks, a free and scalable time series AutoML framework can be developed to perform demand forecasting and revenue optimization at scale.

2.7 Optimization

Optimization techniques have been used in the past for finding optimal pricing strategy. Much of the literature though has focused on discrete optimization-based approaches where the price point could only be set to discrete levels [5]. This may be limiting, especially for larger volume companies such as distributors, since a change of even a few cents may have a substantial impact on the revenue.

The other challenge with optimization techniques is that as the design space becomes non-linear and higher dimensional (i.e. as more products are considered that can cause cross elasticity), optimizers struggle to reach the global minimum [21]. Techniques to aid optimization, specifically for pricing, have been discussed where heuristics are developed in order to simplify the optimization problem [22], [23]. Alternately, this study proposes the use of association mining rule to pick the cross products that are "most influential" in predicting the revenue of another product. Since the number of cross products that are considered can be controlled using this technique, all products do not need to be taken into consideration simultaneously when building the demand models for a single product. This helps reduce the dimensionality of the problem space which assists the optimizer in finding the global minima more easily.

More recently, Bayesian optimization has emerged as a promising technique that allows for a more efficient search of the sample space leading to optimal results [24]. By providing a practical approach to exploitation of the best-known search space and the exploration of unknown search spaces, this approach has been shown to yield faster and better optimization compared to other techniques.

3. Methods

In order to address the research objectives, this study focuses on data for the "Super Premium Whiskey" market from a large distributor of alcoholic beverages. This is a relatively lower volume, high margin business where decreasing the price by a small amount may spur additional demand, increase sales and revenue which could add substantially to the bottom-line.

3.1 Exploratory Data Analysis (EDA)

The data consisted of 201,730 observations of weekly transactions from January 2013 to December 2019. It was broken down by individual customers and products. A total of 278 distinct products from 50 different brands were identified in the data. The majority (215) of these products belonged to the Super Premium Whiskey category.

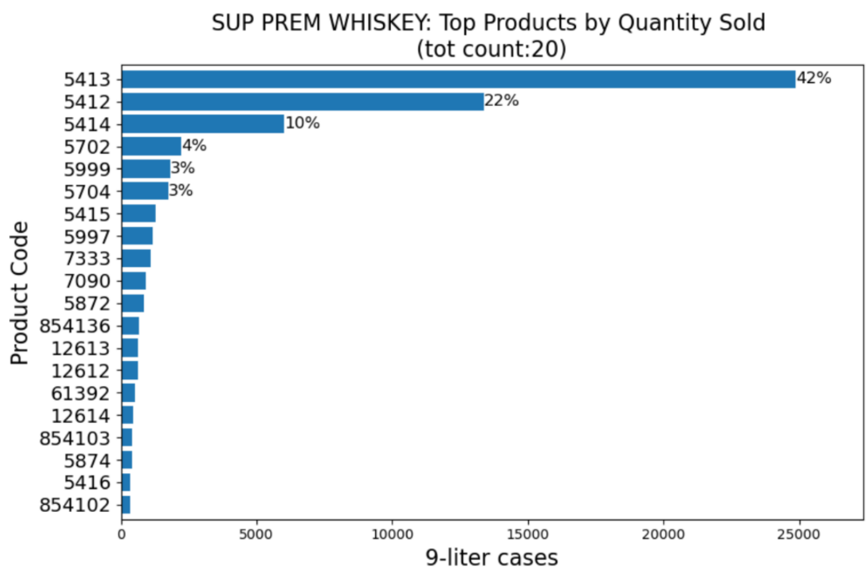


Fig. 1. Distribution of number of transactions vs. 9-liter equivalent cases sold by product in the Super Premium Whiskey. The percentage near the horizontal bar is the proportion vs. the total quantity sold.

An initial analysis of the transactions for the Super Premium Whiskey category shows that the top three products represent a large proportion of the total quantity sold. This is visible in Fig. 1 which shows the quantity sold for each product. This quantity is represented in terms of "9-liter Cases" which is an industry standard measurement term that standardizes the quantity regardless of the size of the bottles in the case (375ml, 750ml, 1L, 1.75L, etc.). This helps in aggregating the quantity across the various transactions. It can be observed that the top three products represent 74% of the total quantity sold. In order to build a more effective revenue optimization model, this work focuses on these top three products and the correlation of their price with the demand for the other top products. It also examines the effect of including varying number of top products (from three to six) and correlated similar products (from one to five) on the overall demand forecasting capability.

Fig. 2 Distribution of price per 9-liter case for the most sold product in Super Premium Whiskey category

A model that takes price elasticity into account needs to leverage the variance in the price at which the product is sold over time and how the sales vary as the price changes. Fig. 2 shows that the variance in the price of a product in the Super Premium Whiskey category. Products in this category have a very high variance (pricing varying from $200-$240 in most cases) which is required to capture any price elasticity effects in the forecast model. Had there been little to no variance, it would have been difficult to capture these trends using the techniques described in this paper.

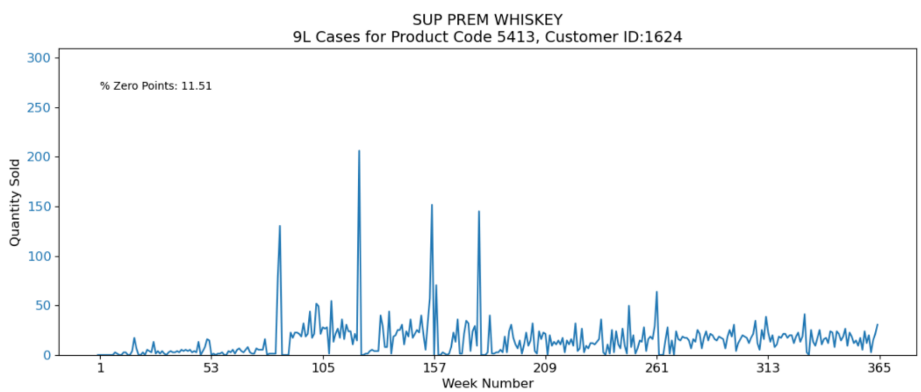


Fig. 3 Quantity sold per week, for the top product in the Super Premium Whiskey category for a Customer.

In the beverage market there is an expectation of seasonal behavior, where some periods have higher sales due to holidays and weather. Fig. 3 shows the quantity sold for the top product in the Super Premium Whiskey category for a customer. It is visible that there are periodic peaks in the quantity sold, and some hints of seasonality. Hence, seasonality is considered a key factor and needs to be incorporated in the models used by this paper. In addition, the top product in the Super Premium Whiskey category shows a bimodal behavior in the quantity sold (Fig. 4) which could be indicative of multiple seasonalities i.e. spike in demand at two different times of the year. Hence, using the Facebook Prophet library approach may be beneficial in capturing these trends since the techniques in this library work well with data having multiple seasonalities.

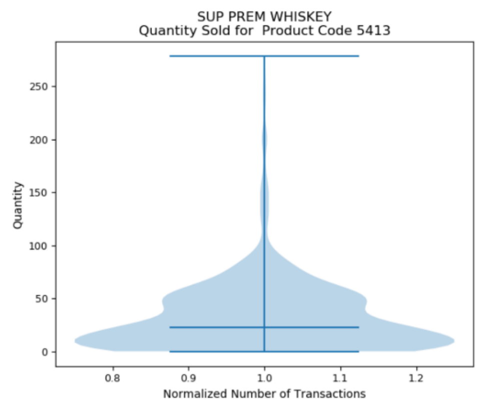


Fig. 4. Distribution of quantity of products sold for the most sold product in the Super Premium Whiskey category.

There is also evidence of outliers in the data that can be caused by human error or other external factors like special sales events. Some of the weeks in Fig. 3 also show zero quantity sold, which can be caused by external factors like data entry errors, closure, or slow-down in the business. This work considers that these data are based on real-word situations where human error, systems update, and record adjustments may add large noise in the data. In order to reduce the noise, the data is resampled at the monthly level which reduces the outliers and periods of low or no purchase. This can be seen from Fig. 5 and Table 1.

Lastly, the exploratory analysis looks at the variation of price over time and how it impacts demand. Fig. 6 shows this behavior for the top product in the Super Premium Whiskey category. There is evidence of fluctuation in the price over time, with some impact on the quantity sold. When the price increases the total quantity sold tends to reduce. Hence, the models in this research uses price as exogenous variable when predicting demand.

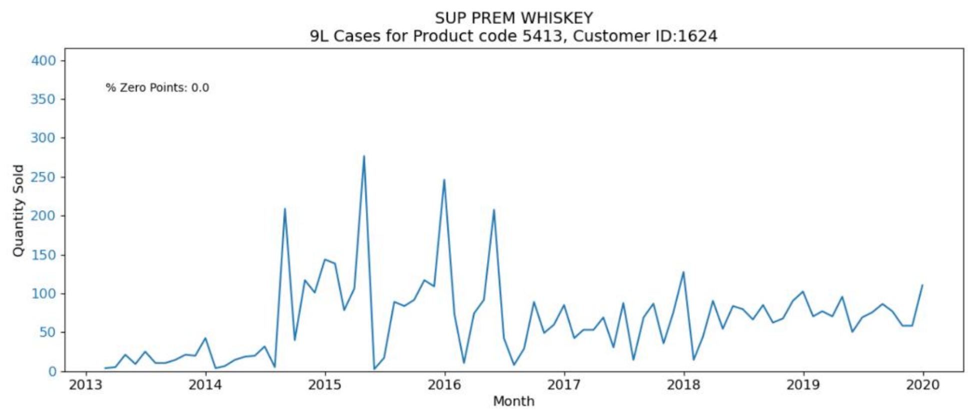


Fig. 5 Quantity sold per month, for the top product in the Super Premium Whiskey category for a Customer

Table 1. Percentage of observations with zero sales (weekly vs. monthly) for the top three Super Premium Whiskey products for a Customer

Top Product Code	% of periods with 0 sales (Weekly)	% of periods with 0 sales (Monthly)
5413	11.51	0
5412	53.15	17.5
5414	48.48	13.41

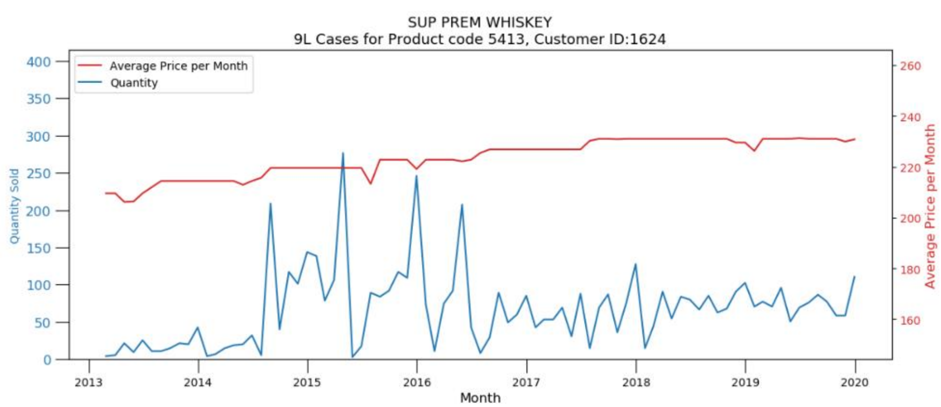


Fig. 6 Quantity sold per month and Price per 9-liter case for the top product in the Super Premium Whiskey category for a Customer.

3.2 Overall Algorithm

The study builds revenue optimization models on a per customer and per product category basis. The overall algorithm flows as shown in Fig. 7. For each customer (line 1) and product category (line 2 – only Super Premium Whiskey in this case), the top ‘N’ products to be analyzed are selected based on revenue (line 3). These ‘N’ products are those that will have the most likely impact on the overall revenue moving forward. Then, ‘K’ demand models are built using various modeling techniques for each of these products (lines 4-6). For building the demand model and in order to consider the price elasticity, ‘M’ most similar products are selected (line 5) and their price is included in the model as exogenous variables along with the price of the product for which the revenue is being modeled (line 6). Finally, the best models are picked from these ‘K’ models using a suitable metric (line 7). In addition, a rolling window is used for model comparison since it offers a more robust estimate of the models forecasting capability by computing “validation scores” across multiple time windows. This ensures that a single “bad” forecast does not eliminate an otherwise good model.

Once a “best” model is chosen for each product, a revenue cost function is computed for the customer and product category considering all ‘N’ products at the same time (line 8). Details on how this cost function is computed is discussed in the subsequent sections. Next, this cost function is used in an optimization algorithm to maximize the overall revenue of each customer and product category. The variables used to optimize the revenue are the price for the ‘N’ products. In other words, optimization computes the maximum revenue achievable by considering the joint price of the top ‘N’ products for each customer and product category combination.

```
1. foreach customer:
2.   foreach product category:
3.     Select: Top'N' products to analyze
4.     foreach product:
5.       Select: The "most similar" 'M' products from the top 'N' products.
6.       Create: 'K' models to forecast demand based on price of product and the 'M' most similar products (exogenous variables)
7.       Select: Best model from the 'K' models based on metric of choice (default: RMSE) and rolling window
8.     Compute: Cost function for Revenue using best demand model for each of the 'N' products and the price of each product.
9.     Optimize: Maximize revenue from customer for the product category based on cost function and price of each product.
```

Fig. 7. Overall Algorithm

For this research, ‘N’ was chosen to between three and six since it accounted for majority of the overall revenue for each category. ‘M’ was chosen to be between one and five i.e. consider one to five similar products (depends on the value of ‘N’) for the baseline revenue optimization model. ‘K’ was chosen to be four for reasons that will be described in the subsequent sections.

3.3 Identifying Products Similarity through Association Analysis to address Cross Effects

To study the impact of cross-effects on price elasticity, the similarity between products contained in the data is investigated. Ideally, all products in a category would be included when modeling the demand (with cross-effects) for any product. This will give the most accurate measure of cross effects. However, with limited amount of data, this approach does not work since lagged variables must also be considered in the demand analysis. Hence, this study narrows the number of influential products used in modeling. This is accomplished with the help of association analysis (co-purchase patterns) and product characteristic similarity metrics such as brand, product name, flavor, and bottle size.

```
1. foreach product in category:
2.   foreach product in category:
3.     Compute: A measure of product value for customer -
                Number of times these products have been purchased together in the same period (normalized).
4.     Compute: A measure of product interchangeability -
                A comparison of the product name similarity using word vectors.
5.     Compute: Composite score (user defined weighted sum of above metrics)
```

Fig. 8. Algorithm to compute the product similarity

The algorithm to compute the product similarity is shown in Fig. 8. The ideal measure of similarity between products should consider the degree to which two products have similar value for the customers, as they will be more likely to switch between these products. The value of a product for a customer can be defined in terms of the purchase frequency of the product within a defined timeframe [25]. A product with higher purchase frequency shows a higher value for the customer. The measure of similarity in value of two products for a customer can be calculated by counting how many times two products were purchased together in the same time period relative to the total number of purchases during that period, i.e. normalized score between zero and one (line 3). A value of zero represents an absence of similarity in the value of two products for the customer, while a value of one represents maximum similarity.

While this co-purchase measure is an effective way to assess the value of the products for the customers, it does not measure product interchangeability. Instead, similar brand and product name increase the customer's perceived similarity between product variants in the same family (the halo effect) [26]. Hence, they are inclined to evaluate these products jointly when making their purchasing decisions. Due to this, a second measure of product similarity is included by comparing word-vectors of the concatenation of products name, brand, flavor, and bottle size (line 4). Word-vectors are a multi-dimensional representation of word meaning wherein two similar words tends to be close to each other in their vector space. This methodology goes beyond the simple syntactic regularities by also considering the semantic representation of the word [27]. With this technique two products with different names but similar characteristics would have a high similarity score. The calculation of this measure has been performed by using the word2vec algorithm, provided by the Python library spaCy [28]. This algorithm returns a similarity score for two words which is between zero and one. Values close to one imply the products are semantically similar and vice versa for values close to zero.

The two measures of similarity, i.e. "product value" and "characteristic similarity", are weighted (line 5) and this composite similarity measure is the final score that represents how much the two products are similar and interchangeable. The weighting is user defined and can be selected by the domain experts so that the similarity results make practical sense.

3.4 Demand Modeling

The models that can be used in this study need to have the ability to pass exogenous variables as inputs to the model. This was needed since in the optimization step, the price was used as the variable for optimization of the revenue.

For the demand modeling, four models were considered for the flow. Because the data showed seasonal behavior, Seasonal ARIMA models with Exogenous variables (SARIMAX) were included in this analysis. A Machine Learning (ML) based approach was also included where the time series problem was converted to a supervised learning problem by considering the previous observations as inputs [29]. This transformed data could then be fed into a plethora of supervised learning ML models and a best model chosen from amongst these ML models. In addition, the techniques described in the Facebook Prophet library were also employed in order to capture the multiple seasonality effects discovered during the exploratory analysis (Fig. 4). Finally, a variant of SARIMAX (Auto ARIMA) which performs automated parameter selection for ARIMA was also used.

From literature it was found that VAR model works well in many cases [3]. Although the VAR models provide an option to include endogenous variables in the data, it does not give control over these variables in the model since this model will automatically and jointly forecast these variables along with the variable of interest. Since the price cannot be controlled directly in the model, this model could not be included in this study.

It was also found [10], [11] that more complex deep learning models were not suitable since they tended to overfit the data. Considering that there were a maximum of 84 observations per product (when resampled monthly), this conclusion would apply in this study as well. Hence, it was decided to exclude these models from this analysis.

Since the study needed to build over 70 models (three customers, up to six products, four models per product), an automated flow was needed. Hence, this research also developed such a system that allows for the user to create all models for a single product using just two lines of code [30]. The complexity of the model building process was encapsulated within the two lines and made transparent from an end user's perspective.

3.5 Model Evaluation

The metric used to evaluate the models was chosen to be Root Mean Squared Error (RMSE). Since the output variable (demand) was numeric, this was an appropriate metric. This metric essentially computes the error of each observation after disregarding the sign of the error – i.e. underprediction and overprediction by the same amount are treated as the same amount of error. Also, since the resulting error is on the same scale as the output variable, it has a human readable meaning to it. Particularly, in this case, an error of 'X' means that the demand predicted was either 'X' above or below the actual demand for that period. This is a useful interpretation compared to metrics such as Mean Squared Error (MSE) where the metric is not on the same scale as the output variable. For example, if demand is in the 100's and RMSE is in the single digits, it could imply that the model is good. It is possible to make these conclusions since the error is on the same scale as the demand. However, with other metrics which are not on the same scale, making this direct comparison becomes difficult.

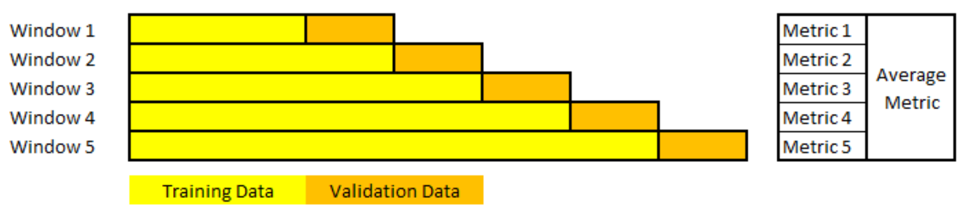


Fig. 9. Rolling Window Forecasts

Another important consideration is the window to use to validate the models. Traditional approaches use a single window of comparison. However, this may not lead to choosing the best model for the data. For example, a window can be chosen where an otherwise good model predicts poorly. This could eliminate a model that may otherwise have generalized well to the prediction of unseen future data. Conversely, if a window is chosen where an otherwise bad model predicts well, the AutoML system may end up choosing a model that will not generalize well to future predictions. Hence a better approach is to use a "rolling window" approach as described in Fig. 9. In this approach, the model is evaluated multiple times using different time periods. The evaluation metric is computed using the validation data from each of these windows and is averaged out to obtain a more robust estimate of the models forecasting capability. It is important to maintain the sequence of the data when creating these windows since the models will still be autoregressive in nature and use previous time points to make future forecasts. Mixing future data observations in the training data would lead to data leakage and the model's performance in production will not be as good as what it was during the training.

3.6 Optimization

Once the models have been developed for a customer and all the products in a certain product category, a maximum revenue can be computed for this customer and product category combination using the joint pricing information for these products. In order to do so, the pricing can be fed into an optimization algorithm along with a cost function that computes the revenue for the customer and product category combination. This study uses Bayesian Optimization since it offers an efficient search mechanism that minimizes the optimization time.

```
1. revenue = 0
2. foreach product:
3.    Compute: mask = False if customer purchases product; True if customer does not purchase product
4.    Compute: revenue += [demand (per best model) * price of product] * !(mask)
5. return -revenue
```

Fig. 10. Algorithm to compute the Cost Function for revenue maximization

The algorithm to compute this cost function is shown in Fig. 10. The algorithm begins by initializing the revenue to 0 (line 1). For each product in the category (line 2), the algorithm decides whether to add the revenue for that product into the cost function or whether to mask it (line 3). This decision is based on whether the customer is going to purchase a product in that time period. This is an important step since this study is not about influencing a customer to purchase a product when they don't have a need for it. Rather it only looks at the optimization of pricing to maximize the revenue after the customer has indicated the willingness to buy the product (i.e. the distributor has received a request for quote or RFQ). Hence, this revenue maximization flow is applicable only when a customer is already purchasing a product.

Once the mask is computed, the algorithm computes the revenue for the product by multiplying the output of the demand model for that product by the price of the product. The resulting revenue is then added to the overall revenue taking mask into consideration (line 4). Once all the individual revenue calculations are added to the overall revenue, this overall revenue is returned as an output of the cost function which can then be used by the optimizer. It is important to note a small subtlety here that the optimizer only minimizes the value returned by the cost function. Since the goal is to maximize the revenue, the cost function returns the negative revenue value and the optimizer minimizes this which is equivalent to maximizing the revenue.

```
1. Define: MARGIN
2. foreach product:
3.    Compute: Min Search Space Value =
               (MARGIN) perentile of the Min product price for that customer product combination
4.    Compute: Max Search Space Value =
               (100-MARGIN) percentile of the Max product price for that customer product combination
```

Fig. 11. Algorithm to compute the Search Space for the Optimizer

The other important consideration for the optimizer is the definition of the search space for the pricing of each product. This confines the minimum and maximum value of the pricing for each product that the optimizer will consider. It is important to note that since the price range of each product will be very different, the search space must be computed on a per product basis. The algorithm to compute this search space is shown in Fig. 11. The algorithm starts with the pricing analyst (or equivalent) defining a “MARGIN” (line 1). This is a parameter that is set based on how high or low the analyst wants the price to go with respect to historical pricing and to remove the impact of outlier price points. Next, the algorithm computes the minimum value of the search space for each product by looking at the MARGIN percentile of the historical price for the product. Similarly, the maximum value of the search space is constructed by looking at the (100-MARGIN) percentile of the historical price for the product.

4. Results

4.1 Top Three Products with 1 Cross Effect

The first step in the algorithm is the computation of product similarity as described in Fig. 8. The results can be visualized using the heatmap in Fig. 12 and using Table 2. This figure shows the similarity matrix for the three most popular products in the Super Premium Whiskey category. The most similar product to Product ID 0 (corresponding to the top Product Code 5413) is Product ID 1 and vice versa. For Product ID 2, the most similar product is Product ID 0. This information is used to build the models for each product (i.e. for Product ID 0, the price of Product ID 0 and the price of Product ID 1 are taken into consideration, etc.).

Table 2. Most Similar products for the top three products in the Super Premium Whiskey category

Top Product Code	Top Product ID	Most Similar Product ID
5413	0	1
5412	1	0
5414	2	0

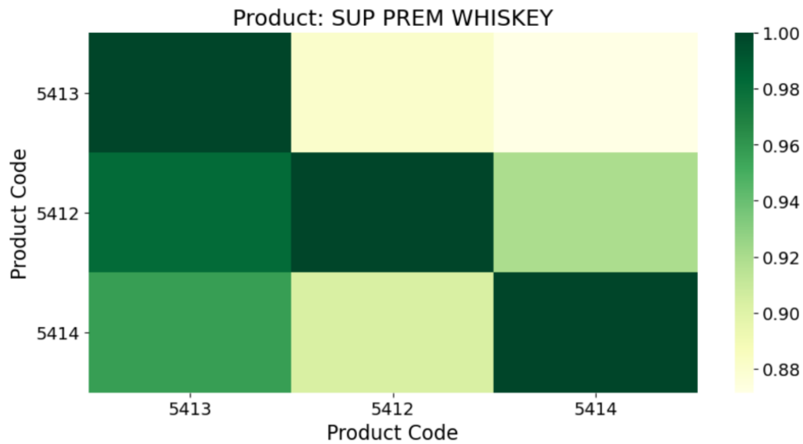


Fig. 12. Similarity matrix for the top three Super Premium Whiskey products.

Table 3 White noise and model selection for top three Super Premium Whiskey products

Product ID	White Noise	Naïve Baseline Model	AutoML (without cross-effect)	AutoML (with 1 cross-effect)
All Customers Combined				
0	No	Last	Auto ARIMA	Auto ARIMA
1	Yes	Mean	SARIMAX	SARIMAX
2	No	Mean	SARIMAX	Auto ARIMA
Customer ID 1				
0	Yes	Mean	SARIMAX	SARIMAX
1	Yes	Mean	SARIMAX	ML
2	Yes	Mean	Auto ARIMA	ML
Customer ID 2				
0	Yes	Mean	ML	ML
1	No	Last	ML	ML
2	No	Mean	ML	ML
Customer ID 3				
0	No	Mean	Auto ARIMA	Prophet
1	No	Mean	ML	Auto ARIMA
2	No	Mean	SARIMAX	Auto ARIMA

Before time series models are constructed for the demand, a test for white noise should be performed to check for the availability of signal in the data. These results are captured in Table 3. Model analysis should only be limited to cases where the data is not consistent with white noise.

Next, naïve baseline models were constructed for each of the products in the Super Premium Whiskey category. These served as a reference for comparing the statistical and machine learning models. Three naïve models were constructed for each product – the first one predicted using the last available demand value, the second one predicted using the average of the last eight months demand value, and the third one predicted using the average of the demand from the same month in the last two years (seasonal period). Then, the best naïve model was selected from these three and treated as the baseline model. Further, 'K' models were constructed as described in the algorithm and the AutoML framework picked the best one based on the RMSE. These results are also captured in Table 3. Fig. 13 shows the impact of cross effects as captured by these models for "All Customers Combined" where the data was not white noise (Product ID 0 on the left and Product ID 2 on the right).

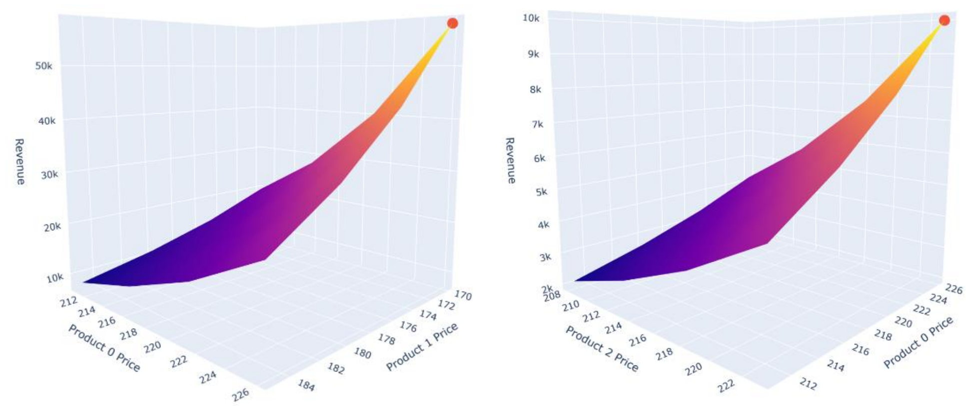


Fig. 13. Impact of Cross-effects on Revenue of Product ID 0 (left) and Product ID 2 (right)

Once the best models have been selected for each product, the composite cost function and the search space are computed for the optimizer as described earlier. A MARGIN of '5' was used in the baseline flow to remove outliers in the pricing data (limiting the pricing search space between the 5th and the 95th percentile). The outliers can have a negative impact on the optimizer since the models will not be very accurate around these price points.

In order to ensure that the improvement in results were reproducible, the process was tested on three consecutive periods as shown in Fig. 14. In the first period, the models were built using the methods descried above using data from January 2013 till September 2019. These models were then used to find the optimal pricing strategy for October 2019. The revenue using the optimal pricing strategy was then compared to the "status quo" revenue generated using the actual pricing at which the company sold the product during that period. This process was then repeated by expanding the period by one month. The models were built using the data from January 2013 till October 2019. These models were used to determine the ideal pricing strategy for November 2019 and the results were compared to the status quo as before. Finally, the period is expanded by one more month (models built using data from January 2013 till November 2019) and the comparison process outlined above was repeated. In the end, the change in the expected revenue vs. the actual revenue from these three periods was aggregated to produce a more robust estimate of the performance of this "optimal pricing" computing engine.

Fig. 14. Testing Methodology for Reproducibility

Table 4 Models' RMSE for top three products in the Super Premium Whiskey category (across all periods and for all customers Combined)

Train Period End	White Noise	RMSE Naïve	RMSE (without cross-effect)	Improvement vs. Naïve	RMSE (with 1 cross-effect)	Improvement vs. Naïve
All Customers Combined | Product ID 0						
09/19	No	0.3192	0.2311	28%	0.1884	41%
10/19	No	0.3239	0.2191	32%	0.1620	50%
11/19	No	0.3245	0.2439	25%	0.1837	43%
All Customers Combined | Product ID 1						
09/19	Yes	0.0339	0.0578	-70%	0.0517	-52%
10/19	Yes	0.0216	0.0346	-61%	0.0310	-44%
11/19	Yes	0.0432	0.0288	33%	0.0282	35%
All Customers Combined | Product ID 2						
09/19	No	0.3731	0.2941	21%	0.2162	42%
10/19	No	0.3732	0.2761	26%	0.2853	24%
11/19	No	0.4280	0.2603	39%	0.2782	35%

Table 4 shows the model performance when three top products were considered. For reasons described in 5.1, Table 4 is limited to models that were generated using data for all the customers combined. In addition to the model with cross-effects, a model without cross-effects was also constructed for comparison.

Fig. 15 shows the price adjustment recommended by the optimizer when the three top products and one cross effect were considered. The "Non Optimized" (dashed) lines show the status quo pricing, which was offered during this period, whereas the solid lines show the pricing recommended by the optimizer. Fig. 16 shows the corresponding change in revenue forecasted due to the adjustment in the pricing. Overall, taking all three products into consideration, the revenue was expected to increase by 29.7% using the optimized price point.

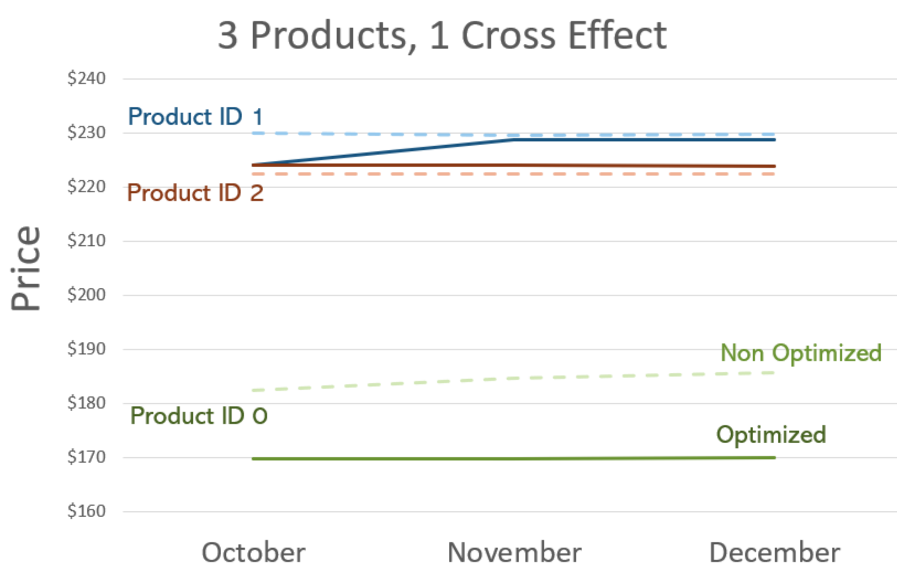


Fig. 15. Price Adjustment Recommended by Optimization Engine

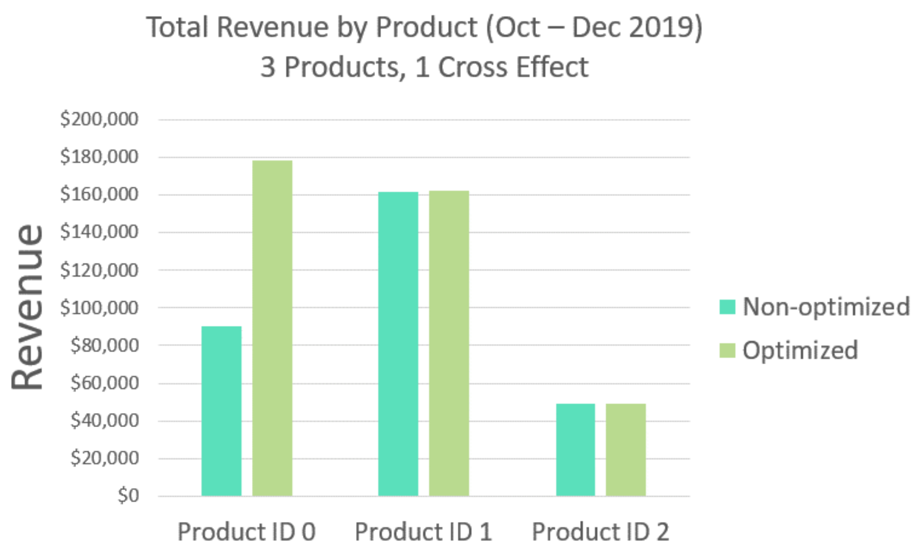


Fig. 16. Predicted Change in Revenue due to Sales at the Optimized Price Point

4.2 Results after Including More Products, Cross Effects, and Evaluation Periods

In this study, the impact of adding more similar products to the model development process was also evaluated. In order to do this, several combinations of top products and similar products (cross-effects) were considered and the performance was evaluated on two vectors. First, the average RMSE of all the models was compared to determine the effect of adding more cross-effects to the model development process. Next, the impact of adding more cross effects was studied on the performance of the optimizer.

The results of the average model RMSE (for all models) for various combinations of top products and cross effects can be seen in Fig. 17. The X-axis represents the number of top products and the Y-axis represents the improvement over the average RMSE of the naïve models. The blue line shows the results for the models without cross-effects (i.e. only considering the price of the product being modeled). The lines in various shades of green represent the results of the flow with different number of cross-effects taken into consideration. It is important to note that when "N" top products were considered, only up to "N-1" cross effects could be chosen.

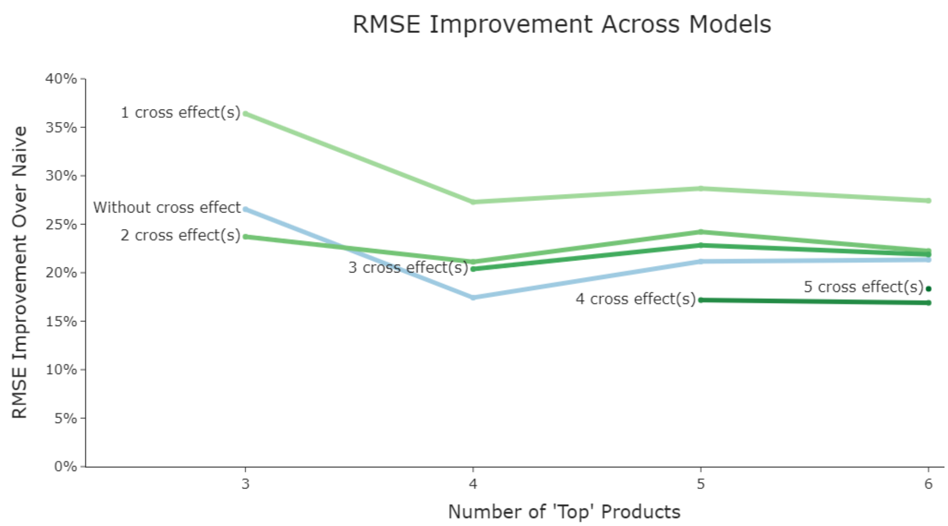


Fig. 17. Impact of Adding More Top Products and Cross-Effects on Model Performance

 # Similar Products (Cross Effects)

# Top Products	1	2	3	4	5
3	29.7%	14.8%			
4	29.6%	15.6%	17.3%		
5	24.3%	12.9%	18.2%	42.5%	
6	26.5%	22.0%	7.7%	44.8%	14.8%

Fig. 18. Impact of Adding More Top Products and Cross-Effects on Optimizer Performance

Fig. 18 shows the improvement in revenue forecasted for various combinations of top products and cross effects as a result of change in pricing recommended by the optimizer. The result corresponding to three top products and one cross-effect matches the result shown in Fig. 16. The other values were computed using a similar methodology as the one used for three top products and one cross-effect.

5. Discussion

5.1 Model Segmentation

The models developed in this study were initially segmented based on product category and individual customers. This was in line with the literature survey which indicated that individual customer buying patterns might have an influence in improving the accuracy of the final prediction. However, on careful evaluation of this data, it was observed that when looking at individual customers, much of the demand was consistent with white noise as is shown in Table 3. A Ljung-Box test was conducted with lag $K = 24$ and 48 on the demand data to test for white noise. In many of the cases, the test failed to reject the null hypothesis (data is white noise). Hence even though modeling at the individual customer level would have resulted in capturing the individual customer buying patterns, the final models were developed at the product category level only (with demand aggregated across all customers in that category). Though still not perfect (there were still some cases where the data was white noise), this resulted in more signal in the demand data in general which could then be modeled using appropriate modeling techniques.

5.2 Impact of Adding Cross-Effects on Performance

Using the Super Premium Whiskey category as reference, it can be seen that the models with cross effects tend to perform significantly better than the naïve models and marginally better than the models without cross effects (Table 4). Specifically, when compared to the naïve models, the RMSE of the models without cross effects improved by an average of 29% for the products where demand was not white noise (Product ID 0 and 2). For these same products, the model with cross effects improved the RMSE by an average of 39%. When compared to the models without cross effects, the models with cross effects improved the performance by 14%. These results show a significant improvement over the naïve modeling approach as well as modeling without cross effects. They also correlate well with the literature survey which indicated that the cross effects were an important phenomenon to consider when modeling demand.

Another significant finding is the impact of pricing on the revenue as shown in Fig. 13. Again, only products where the demand is not white noise are considered for this analysis. For both Product ID 0 (left) and Product ID 2 (right), as its price of the product increases, the revenue increases (the model predicts that the increase in the price of the product can make up for the drop in its demand). What is also interesting to note is the impact of the similar products. For Product ID 2, as the price of the similar Product ID 0 increases, the revenue shifts towards Product ID 2. This is consistent with the cannibalization findings in the literature. However, for Product ID 0, as the price of the similar product (ID 1) is decreased, the revenue for Product ID 0 increases. This is contrary to the literature survey and it may point to other confounding factors that may not have been considered while developing the model.

When the output of these models is fed to the global optimizer, it shows some noteworthy results. Since there is no practical way to test the efficacy of the pricing optimization process without a A/B test in the field, the expected revenue using the optimized pricing was compared to a “what-if” scenario using the “status quo” pricing (the price at which the product was actually sold during the evaluation period) instead. The suggested results from the optimizer for the top three products in the Super Premium Whiskey category (Fig. 15) seems to indicate that the price of Product ID 0 should be reduced for all three periods (October, November and December) from about $185 down to about $170. The price of Product ID 1 should be reduced in October from $230 down to $224 but can be increased back up close to $230 during the holiday season of November and December. For Product ID 3, the optimizer recommended increasing the pricing slightly from $223 to $224. This combination of price changes would lead to the highest increase in the revenue as predicted by the optimizer.

Since the most drastic impact was made to the price of Product ID 0, we would expect the most drastic impact to its revenue as well. This effect can be seen clearly in Fig. 16. The revenue for Product ID 0 is expected to increase substantially (from $90,240 to $178,456), while the revenue for Product ID 1 (from $161,504 to $162,706) and Product ID 2 (from $48,937 to $48,861) is expected to remain roughly the same despite the price changes. In all, the overall revenue is expected to increase by 29.7% using the optimized pricing.

5.3 Impact of Including More Products and Cross Effects on Performance

Fig. 17 shows the impact of adding more top products and cross effects into the optimal pricing framework described in this paper. Although all combinations provided an improvement over the naïve models, the best results were obtained when only one cross effect was included. In all cases, adding more cross effects seems to deteriorate the forecasting performance of the models. This can be attributed to the limited data available to develop the models. Since the data was resampled at a monthly frequency, there were a maximum of 84 observations available to build each model. Adding more cross effects left less degrees of freedom available to determine the model parameters. This was because the model had to include not only the price of the cross-effects, but also their lagged values. This lowering in the degrees of freedom led to an overall lowering in the performance of these models. In fact, addition of more than three cross effects led to a performance degradation below the level observed for the models without cross effects.

Another significant finding was that the performance of the system remained stable as more top products were added to the framework. It can be seen from Fig. 17, that for the same number of cross effects, as more products were added, the RMSE improvement remained consistent. This points to the fact that framework is scalable and that more products can be added to it without concern.

Similar effects can be seen in Fig. 18, which shows the performance of the optimizer. Again, the best results were generally obtained when only one cross effect was included, and the results began to deteriorate as more cross-effects were added (except for four cross-effects). Also, like the results for the model RMSE, the performance of the optimizer remains stable as more top products are added (for the same number of cross effects).

In summary, the pricing optimization system seemed to be stable and could support the addition of more top products if needed. Including only one cross-effect seemed to be the most optimal in this scenario due to the limited number of observations available to build the models. This could however be different in situations where more data is available and should be evaluated using a similar methodology as described in this study.

5.4 Limitations

As much as the results from this study are enticing for pricing managers, they are not without their own limitations. For example, there could be a lot of factors that could impact the demand for a product beyond just its price (which are unaccounted for in the models). These could include customer inventory, state of the economy, consumer behavior, effect of advertisement and other marketing efforts by the distributor. This can be seen in Fig. 13 where the impact of reduction in the pricing in Product ID 1 on the revenue for Product ID 0 could not be explained (left figure). In addition, while there is an expectation of a large signal to noise ratio in the demand data, due to the interaction of all these effects, the resultant data may be choppy and may be more consistent with white noise as was observed in many of the cases.

Finally, while the models do capture the cannibalization and halo effects, ultimately the true interaction pattern is best captured with experimentation. Hence, a randomized A/B pilot study would be recommended as a follow-up to determine the true efficacy of the optimization process. This would also help to determine if the change in price is truly causing a change in demand (taking other factors into consideration) vs. simply being correlated to it and confounded by other factors.

5.5 Legal and Ethical Considerations

The legality of customizing prices to the individual buying entity, in this case store chains, needs to be considered. Dynamic pricing is legal so long as sellers aren't biased based on a customer's race, religion, nationality, or gender [31]. However, the buyers in this study are store chains which don't necessarily have any of these characteristics. Despite that, an argument could be made that these stores are owned by or cater to those primarily of a single demographic. Due to this it must be noted, this research should not be implemented based on discriminatory practices and instead should be based purely on buying propensity of the customers (data driven based on historical behavior).

Though dynamic pricing based on data is currently legal, it doesn't necessarily mean that it's ethical. In the current era where algorithmic methods allow for dynamic pricing at scale, there needs to be greater discourse about the impact of these decisions on the market and the everyday consumer. This study aims to be fair and equitable to all people. The lack of transparency in model decisions regarding pricing allow for the possibility of price reflecting demography for consumers. There is still the possibility that vulnerable demographics will be penalized due to pricing. However, since this model indirectly aims to capture what customers are willing to spend it may serve as a tool for equalizing the relative cost of the product. For instance, affluent customers may be willing to pay more so their pricing may be adjusted accordingly. Further research is needed to see if this algorithm does raise prices for chains catering to more affluent demographics however it may very well be the case.

6. Conclusion

The purpose of this study was to develop a scalable revenue maximization system for distribution companies of non-perishable products. The data for the analysis were seven years of time series records of price and sales for a distributor of alcoholic products. Models were developed to maximize the revenue for the Super Premium Whiskey category of products. Top three products were chosen from this category since they accounted for most of the revenue.

Based on the literature survey, cross effects such as cannibalization, and halo effects were included in the model by including the price of similar products. Since models had to be developed for multiple products, an automated machine learning system was developed to create multiple models for each product (SARIMAX, Auto ARIMA, Machine Learning, Prophet) and the best one was chosen based on the rolling window RMSE. The models developed by this automated system were compared to naïve models and models without cross effects. It was found that the models with cross effects improved prediction performance by 39% over the naïve models whereas the models without cross effects only improved the performance by 29%. Once the models were selected, revenue maximization was performed using Bayesian Optimization. The results of the optimization indicated that optimized pricing could potentially increase the revenue by 29.7%.

The impact of adding more top products was also studied to ensure scalability. The results showed that the performance of the system remained consistent even when six top products were considered. This points to the stability and scalability of this approach. Finally, the impact of adding more cross effects was also considered. Although it might be intuitive to think that adding more cross effects would lead to better performance, the limited data to build the models meant that adding one cross effect led to the best performance and the performance deteriorated when more cross effects were added.

Some of the issues found during this study had to do with the noise in the data. Segmentation at the customer level would have captured the individual buying patterns, but the data was too noisy (consistent with white noise in many cases). Hence the data had to be aggregated to include all the customers in the product category.

There are several different avenues for future work with the methodology developed in this study. Confounding factors such as geographic and economic indicators, social network data, marketing effects, weather, and political factors were not considered in this study. Addressing the cold start problem posed by the COVID-19 pandemic could also be included. Also, being an observational study, conclusions about causation between price and demand cannot be drawn. Hence, the suitability of this study should be conducted using a randomized A/B test.

Finally, the methods proposed in this study are generic and scalable in nature and not specific to the distribution industry or the alcoholic beverage market. Hence, they can be applied to other sectors as well and could have a large impact on earnings and the general approach to pricing.

7. References

1. OpenStax. (2016, May 18). *Principles of Economics.* Retrieved from BC Open Textbooks: https://opentextbc.ca/principlesofeconomics
2. Bromley, R. (2006). *Elasticity*. Retrieved from pcecon.com class notes: https://www.pcecon.com/notes/elasticity.html
3. McColl, R., Macgilchrist, R., & Rafiq, S. (2020). Estimating cannibalizing effects of sales promotions: The impact of price cuts and store type. *Journal of Retailing and Consumer Services, 53.*
4. Meredith, L., & Maki, D. (2001). Product cannibalization and the role of prices. *Applied Economics*, 1785-1793.
5. Bayoumi, A.-M., Saleh, M., Atiya, A., Aziz, H., & Atiya, A. (2013). Dynamic Pricing for Hotel Revenue Management Using Price Multipliers. *Journal of Revenue and Pricing Management, 12*(3), 271-285.
6. Kotler, P., & Keller, K. (2012). *Marketing Management.* Boston: Pearson.
7. Kerin, R. A., Harvey, M. G., & Rothe, J. T. (1978). Cannibalism and new product development. *Business Horizons, 21.5*, 25-31.
8. Pickett, J. C., & Reilly, D. P. (2020, May). *Testing Marketing Hypothesis.* Retrieved from Automatic Forecasting Systems: https://www.autobox.com/pdfs/cannibalize.pdf
9. Akçay, Y., Natarajan, H., & Xu, S. H. (2010). Joint Dynamic Pricing of Multiple Perishable Products Under Consumer Choice. *Management Science, 56*(8), 1345-1361.
10. Jiang, L., Rollins, K. M., Ludlow, M., & Sadler, B. (2020). Demand Forecasting for Alcoholic Beverage Distribution. *SMJ Data Science Review*.
11. Arora, T., Chandna, R., Conant, S., Sadler, B., & Slater, R. (2020). Demand Forecasting In Wholesale Alcohol Distribution: An Ensemble Approach. *SMU Data Science Review*.

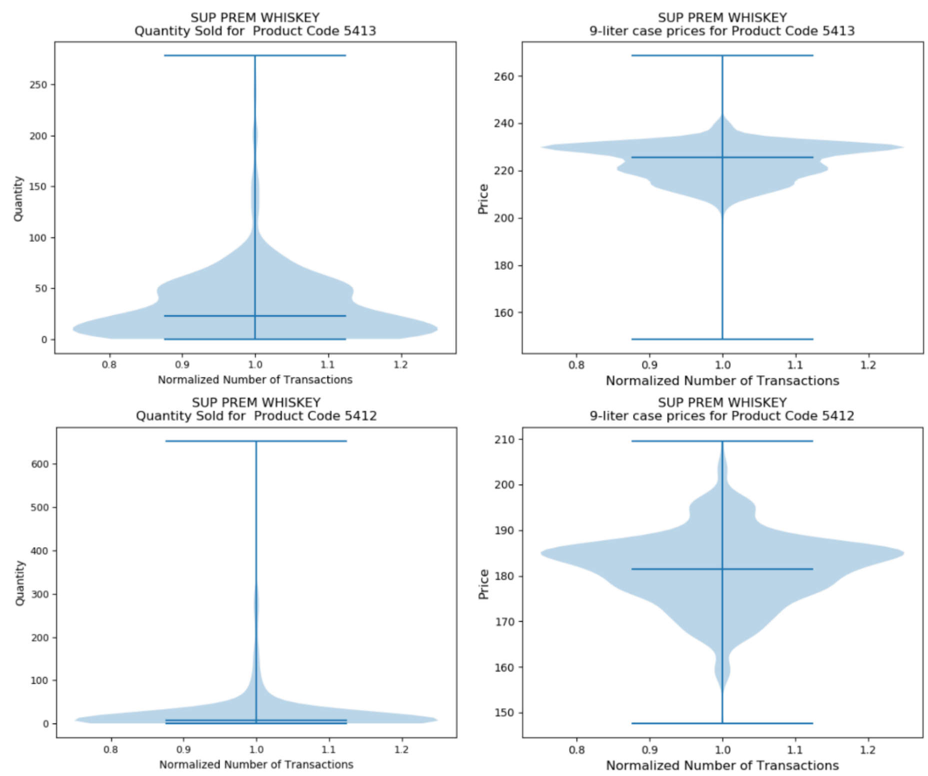
12. Chevalier, M. (1975). Increase in Sales Due to In-Store Display. *Journal of Marketing research, 12*(4), 426-431.
13. Blattberg , R. C., & Wisniewski, K. J. (1989). Price-Induced Patterns of Competition. *Marketing Science, 8*(4), 291-309.
14. Neslin, S. A. (2002). *Sales Promotion, Relevant Knowledge Series.* Marketing Science Institute.
15. Taylor, S. J., & Letham, B. (2017). Forecasting at scale. PerrJ Preprints(5:e3190v2). Retrieved from https://peerj.com/preprints/3190/.
16. Machine Learning Professorship Freiburg. (2019). auto-sklearn. Retrieved from auto-sklearn: https://automl.github.io/auto-sklearn/master/index.html#
17. Rosenthal, E. (2018, MArch 22). *Time Series for scikit-learn People (Part II): Autoregressive Forecasting Pipelines.* Retrieved from Ethan Rosenthal : https://www.ethanrosenthal.com/2018/03/22/time-series-for-scikit-learn-people-part2/
18. Moez, A. (2020, 05). *PyCaret*. Retrieved from PyCaret: https://pycaret.org/
19. H2O.ai. (2018, June 12). *H2O.ai.* Retrieved from Time is Money! Automate Your Time-Series Forecasts with Driverless AI: https://www.h2o.ai/blog/time-money-automate-time-series-forecasts-driverless-ai/ .
20. Perktold, J., Seabold, S., & Taylor, J. (2020, Feb 21). *Time Series Analysis.* Retrieved from statsmodels v.0.11.1: https://www.statsmodels.org/stable/tsa.html .
21. scikit-learn developers. (2019). *Developing scikit-learn estimators*. Retrieved from Scikit Learn: https://scikit-learn.org/stable/developers/develop.html .
22. Dauphin, Y. N., Pascanu, R., Gulcehre, C., Cho, K., Ganguli, S., & Bengio, Y. (2014). Identifying and attacking the saddle point problem in high-dimensional non-convex optimization. arXiv. Retrieved from https://arxiv.org/abs/1406.2572 .
23. Zhang, D., & Cooper, W. L. (2009). Pricing substitutable flights in airline revenue management. *European Journal of Operational Research, 197*(3), 848-861.
24. Gallego, G., & van Ryzin, G. (1994). Optimal Dynamic Pricing of Inventories with Stochastic Demand over Finite Horizons. *Management Science, 40*(8), 999-1020.
25. Shahriari, B., Swersky, K., Wang, Z., Adams, R. P., & De Freitas, N. (2016). Taking the Human Out of the Loop: A Review of Bayesian Optimization. *IEEE*, 148-175.
26. Lui, D.-R., & Shih, T.-T. (2005). Hybrid approaches to product recommendation based on customer lifetime value and purchase preferences. *Journal of Systems and Software, 77*(2), 181-191.
27. Hui, K.-L. (2004). Product Variety Under Brand Influence: An Empirical Investigation of Personal Computer Dema. Management Science, 50(5), 686-700.
28. Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013). Efficient Estimation of Word Representations in Vector Space. arXiv. Retrieved from https://arxiv.org/abs /1301.3781.
29. Explosion.ai. (2020). *Word Vectors and Semantic Similarity.* Retrieved June 10, 2020, from https://spacy.io/usage/vectors-similarity .
30. Brownlee , J. (2017, May 8). How to Convert a Time Series to a Supervised Learning Problem in Python. (Machine Learning Mastery) Retrieved July 10, 2020, from

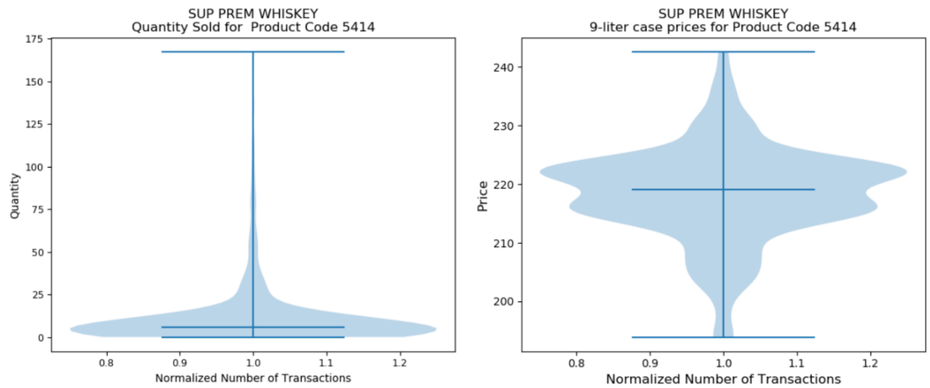
https://machinelearningmastery.com/convert-time-series-supervised-learning-problem-python/ .

31. Seshadri, R., Chen, K., & Gupta, N. (2020, July). Auto-TS. (Auto-TS) Retrieved July 2020, from https://github.com/AutoViML/Auto_TS .
32. Shpanya, A. (2014, January 3). What is price discrimination and is it ethical? (Econsultancy) Retrieved August 9, 2020, from https://econsultancy.com/what-is-price-discrimination-and-is-it-ethical/ .

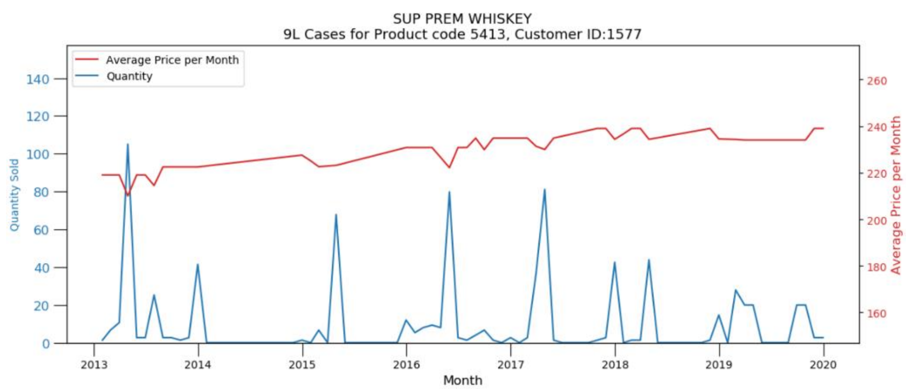
8. Appendix A

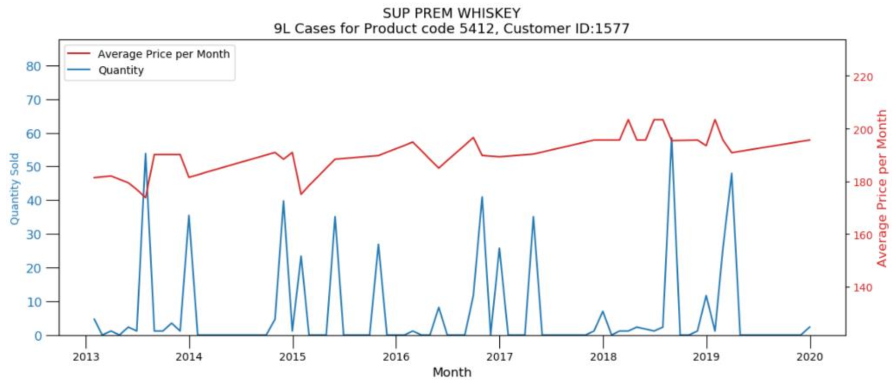
8.1 Distribution of quantity sold and price for top products and top customers.

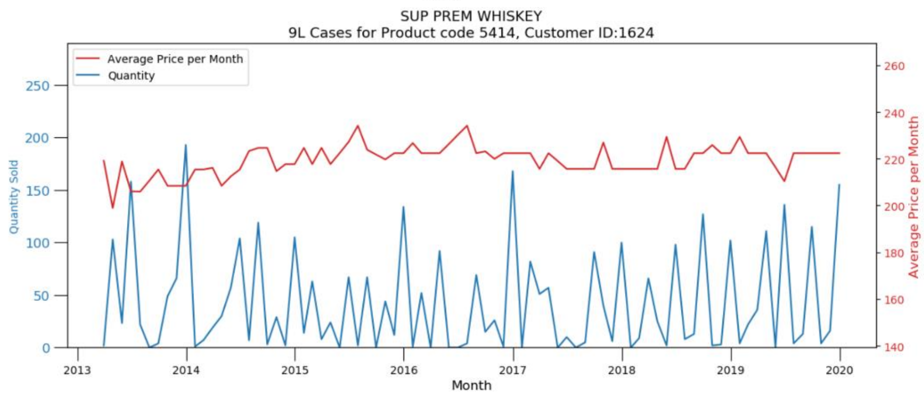




8.2 Distribution of quantity sold and price over time for the top products and top customers




SUP PREM WHISKEY
9L Cases for Product code 5412, Customer ID:1577
Average Price per Month
Quantity
Quantity Sold
Average Price per Month
Month


SUP PREM WHISKEY
9L Cases for Product code 5414, Customer ID:1624
Average Price per Month
Quantity
Quantity Sold
Average Price per Month
Month