UNIVERSITAT DE BARCELONA

FUNDAMENTAL PRINCIPLES OF DATA SCIENCE MASTER'S THESIS

Demand modelling and prediction in retail

Author:
Jael FREIXANET

Supervisor:
Dr. Jerónimo HERNÁNDEZ

A thesis submitted in partial fulfillment of the requirements for the degree of MSc in Fundamental Principles of Data Science

in the

Facultat de Matemàtiques i Informàtica

June 29, 2022

UNIVERSITAT DE BARCELONA

Abstract

Facultat de Matemàtiques i Informàtica

MSc

Demand modelling and prediction in retail

by Jael FREIXANET

Demand forecasting is an indispensable process in the retail industry, since it leads the way to better decision-making for companies. Within this study we compare the results obtained after applying two different machine learning proposals to a case study with real data from a retail company. First, a hypothesis-driven approach where we use a custom model that incorporates econometric knowledge. For that, we combine machine learning methods with a double-logarithmic demand model. Second, a data-driven approach where we use gradient boosting on decision trees to predict demand. Apart from prediction activities, we also revise our results trying to find the optimal price through optimization. The study allows us to reveal the advantages of gradient boosting on decision trees over our custom model, as well as its limitations during the optimization part.

Acknowledgements

First of all, I would like to thank my tutor Jerónimo Hernández for the guidance during the whole process and for helping me to understand that null results are also knowledge and how they can contribute to the research community. Also, thank Theodoros Lappas and Satalia for the advice, enthusiasm and data kindly provided. Thank Jack for the time invested in teaching me ROVA insights. Lastly, thank my family, especially Jan who has helped me and bore with me in good and stressful times.

Contents

Chapter 1

Introduction

1.1 Motivation

Demand forecasting is traditionally done manually and it requires a high number of resources in order to study and predict future sales. Additionally, the retail industry deals with large amounts of data, composed of hundreds of products, for different markets and following different dynamics. Hence, discovering the patterns within previous sales and simulating the future is not a simple task that an employee can accurately do on a recurrent basis. Machine learning gives us the opportunity to overcome all those obstacles in an efficient way. Specifically, machine learning allows us to learn different custom models capable of incorporating additional knowledge hidden in the data. In this direction, we propose a hypothesis-driven study where econometric relations between demand and price are used in our machine learning algorithm.

Demand forecasting is becoming an essential process in retail companies, since their strategic and operational plan is usually formulated around it. Sales prediction helps the business to make better-informed decisions that impact profit margins, cash flow, allocation of resources, opportunities for expansion, stock accounting, operating costs, staffing, price strategy design and overall spend.

From a customer perspective it can provide the client a better experience and fulfill their expectations assuring the stock of the products accordingly. Additionally, demand forecasting allows marketers and businesses to plan offers, launch new products, how or when is better to run flash sales and create a price optimization strategy to maximize benefit.

Summarising, understanding the market and potential opportunities allows businesses to grow, formulate competitive pricing, employ the right marketing strategies, and invest in their growth (Lopienski, 2019) and machine learning can help with this task.

1.2 Industry knowledge

Our case of study is focused on H[1], a retail company based in the United Kingdom (UK), specialized in shoes and shoe care products. Their products are sold in the UK online and offline. It also has online markets in Europe and US with a wide target audience, mainly focused on women and men over 30 years from middle social class.

Shoes are fungible, non-perishable goods but their availability has stock limitations. For instance, a shoe model might be available in all the sizes but not in the most popular one. Some product lines have a limited lifecycle, meaning if the product does not sell as expected or needs to be enhanced the product is discontinued

[1]Company's name not given for confidentiality reasons.

and a replacement might or not be set for the next season. Other products have a long lifecycle due to their characteristics (for example some shoe care products not affected by trends), their design (basic products that will be always purchased) or their sales success. Seasonality is another important factor. Same product (or product line) might be launched for different seasonalities and they can coexist. The product is essentially the same but with some variations on it. We can distinguish two main seasonalities: Spring and Summer (SS) and Autumn and Winter (AW), but we can also find some mid-season launches in some cases.

Promotions and discounts can be applied to specific products or categories but also to the full catalog with the use of coupons or flash campaigns. There are two main sales periods at the end of each seasonality and an outlet section with spare products and old models at a reduced price during all the year.

Advertisement is strongly correlated to the overall demand of the brand but can also be product-focused. In this second case both, product and brand, can be positively affected.

H has a current manual price strategy where costs, competitors, inventory and margin are taken into account. Despite this, decisions are not data-driven but based in their industry acquired knowledge and expertise.

If we approach demand forecasting from the price strategy perspective we need to take into account how often the company is capable of changing prices and when it is recommended to do so. H needs about a week for collecting sales of the previous week and perform all the price changes on their catalogs and databases. This prevents from using sales data from the previous week for predicting current week's sales. Even though this is the capacity of the company, the user experience of the consumers and their price perception should be carefully regarded. Frequently changing the price of the products might be harmful for the brand image.

Due to the sensitive relation of sales with revenue, one of the relevant topics when dealing with the retail industry is to justify why the algorithm recommends to increase or decrease the price. Advanced machine learning algorithms are relegated due to their incapability of satisfying the level of explainability the companies require. That is one of the reasons why linear regression ends up being used in most of the cases in this industry (Antipov and Pokryshevskaya, 2020).

Our study is motivated by the econometric relation between demand and price known as double-logarithmic demand model (Alston, Chalfant, and Piggott, 2002):

$$D = \alpha P^{\beta} \tag{1.1}$$

or

$$\log D = C + \beta \log P \tag{1.2}$$

Where D is the demand, P the price of the product, C is a constant and β is an important parameter called elasticity of demand. The elasticity informs about the responsiveness of the demand to price changes and is negative in most of the cases (Marshall, 1949).

The relation in Equation 1.1 will be the base of our study where we will combine the historical data information, the powerful machine learning methods and the econometric rules based on human behaviour.

1.3 Problem Statement

Our main goal is to be able to predict the demand for calculating the optimal price that maximises revenue. Two main approaches will be followed: a hypothesis-driven study based on the relation between demand and price of Equation 1.1, and a data-driven study using gradient boosting on decision trees and linear regression.

For that, we first built a demand prediction algorithm that tries to fit the historical data provided into a double-logarithmic model where demand has a strong dependence on the price.

Once we are able to predict the demand for a certain product, channel and date at different prices we can compose a demand matrix. Demand matrices are used for advanced price optimizations algorithms. Consists in a n-dimensional matrix containing the demand values obtained for the different feature combinations. For example we could have a demand matrix showing the different demands depending on the date and the price of the product.

Finally, we apply a basic optimization algorithm that calculates the maximum benefit for a price and the corresponding predicted sales.

1.4 Approach

Even though we have enough data for performing some experiments we do not have available data regarding advertisement, promotions, stock or fix cost information (unlike the company experts that acquired it through their experience). In addition, not all the models we will use in this study can manage all the data. This forces us to rethink what type of models and features we will use in each case.

This study explores 3 different approaches to this problem:

- Custom Model: is the main model and is based on the Equation 1.1. This model is sensitive to the number of variables. For this reason we will create product level models for a specific channel where a limited number of features will be used. We will take 12 of the products with more data and apply the different experiments to them.
- CatBoost Model: is a gradient boosting algorithm that will be presented as an alternative to the Custom Model and it will be used to compare the results obtained. In this case we will create a model per category cluster (explained in Section 2.5). The potential of this and the previous Custom Model will be shown by means of a comparison using the Footwear Cleaner product, which is well fitted by both methods.
- ROVA[2] tool model: is based on linear regression used for comparing traditional approaches to our CatBoost model. This model will be the only one using data related to specific holidays. In this last model we have created category level models and their potential will be shown with Womens Goretex category since it is a good representative of both methods.

The price optimization in our case will only take into account the demand matrix with price, and the variable cost related to the production since we do not have advertisement spent, stock or fixed cost information.

[2]tool used for several retail companies and facilitated by Gain Theory.

1.5 Related Work

Machine learning in demand forecasting has been studied and investigated from different perspectives. For example, ANNs have been largely used as a machine learning model for time series forecasting. In (Alon, Qi, and Sadowski, 2001) we can see a comparison between ANNs and traditional statistical methods. Within (Huber and Stuckenschmidt, 2020) they compare a gradient-boosted decision trees algorithm with neural networks for predicting the demand of a bakery on special days. They treat it as a regression and as a classification problem.

In (Smirnov and Sudakov, 2021) they investigate how to deal with new products without historical data using gradient boosting algorithms. Clustering methods for improving performance have been used in (Lingelbach et al., 2021) in the feature generation and (Thomassey and Fiordaliso, 2006) where the clustering procedure carries out groups of similar items in term of sales profile while the decision tree finds understandable links between these clusters and descriptive criteria.

(Ulrich et al., 2022) proposes an automated model selection framework for retail demand forecasting. They consider model selection as a classification problem, where classes correspond to the different models available for forecasting.

Other works try to efficiently solve the high dimensionality of the variable space used in this type of predictions for retails companies. For example, (Ma, Fildes, and Huang, 2016) investigates the value of both intra- and inter-category at product level (which are the relations between products of the same category and products of different categories).

Other studies are using explainability to improve their results as in (Antipov and Pokryshevskaya, 2020), where Gradient Boosting Machines, Random Forests and Elastic nets are compared, and model-agnostic interpretable machine learning techniques (Shapley values) are used for the feature selection in high-dimensional data.

Most of the work done so far does not use inventory (goods in stock) as a driver for improving forecasts as stated on (Fildes, Ma, and Kolassa, 2019), but there are some exceptions. For example in (Tian, Wang, and E, 2021) where they try to study forecasting of intermittent demand using a Markov-combined method that takes into account the inventory status.

Chapter 2

Data Analysis

Our objective is to predict sales in the case study proposed by H company. However, to try to improve the results we will consider several data sources, not only the one provided by the company. Most of them can be found in the repository associated to this study [1]. The data used in this study comes from different sources:

- H Dataset: is our main dataset provided by the company [2].
- Holidays Dataset: containing all the holiday dates in the UK. The data has been merged into our main dataset as a boolean field for each of the holidays.
- Currency Rate Dataset: contains historical exchange rate from dollars and euros to pounds [3] and it is used for unifying the currencies of diverse fields.
- Google Trends Dataset: data regarding the search requests on Google through time in the UK. Datasets can be found on the repository or reproduced online[4].

2.1 H Dataset

This dataset was provided by the retail company and contains the weekly sales information for the last 5 years (from 2017 to 2021). We have around 300.000 rows of data and each row in our dataset describes sales of a product at a specific price, sold via a specific channel, for a concrete discount and at a certain cost. This means that if a product has been sold at two different prices within the same week in the same channel we will have two different rows with the corresponding sales and prices.

Apart from sales, product identifier, price and category, it also includes information related to the Seasonality, the Cost of the product, the RRP (recommended price suggested by the manufacturer), the Markdown (amount by which the RRP is changed), Discounts, Margin and other product attributes. The variable Markdown is a calculated feature that comes from:

$$Markdown = RRP - Price - Discount \tag{2.1}$$

A positive Markdown means the price has been sold for a lower price than the recommended by the manufacturer. More details on the features and their description can be found in Appendix A.

The data has been preprocessed, cleaned and several features have been added as it will be explained in the following sections.

[1] Repository can be found in https://github.com/jaelfv/TFM_2022

[2] Data facilitated by the company. It has not been included in the repository for confidentiality reasons.

[3] Datasets obtained from https://www.macrotrends.net/

[4] Datasets obtained from https://trends.google.com/trends/?geo=UK

2.1.1 Data Cleaning

During the preprocessing of the data several tasks have been performed in order to work with a clean and coherent dataset. There are several fields containing aggregated data that have been transformed into their unitary values. Hence, we have divided the current values per the amount of sales of that row in order to get the unitary value of Price, RRP, Markdown, Margin, Cost and Discount. We have also used the Currency Rate dataset for transforming the dollars and euros to Pound sterling.

Month feature has been eliminated since all the values were empty and the information is already included when combining Year and Week. Season Report has been dismissed since it only gave information about which report the data has been internally stored and coincides with the Seasonality field.

The features Limited Editions Styles (contains information regarding if the product is a limited edition or not) and H Originals (attribute of the product informing if it is an original product of the brand) have been converted into boolean fields where empty values have been taken as False. In the case of Source (informs about the origin of the product) we have created a new value called *NONE* representing all the products where the source was not specified.

In addition to these modifications we have also corrected some typographical errors that were duplicating some categories. For instance, we had Womens Goretex and Womens goretex.

Finally, we have identified and removed some outliers that were showing odd behaviour. For instance, we have removed rows with negative sales, price, RRP and cost as well as markdowns and discounts lower or higher than 400 and -400 respectively.

2.1.2 Additional Features

On one hand, the dataset has been nurtured with new features obtained from already existing data:

- Date: contains the date of the purchase and it is created from the weekly and year information. We have taken as a date representative the Monday of each week. It has been mainly used for visualization, for comparing different methods and for merging Holidays Currency rate and Google Trends datasets.
- Style and gender: obtained based on the category the product belongs to. Gender can take the values Women, Men and Style can take boots, sandals, slippers and goretex values.
- Previous sales and previous mean price: lag variable and average movement variable combined showing the sales from the previous weeks (prev_sales, prev_sales2, prev_sales3 and prev_sales4). And lag variable showing the price of the previous week (prev_mean_price).
- Week_cos and Week_sin transformations: we have used the sinus and cosinus functions in order to represent the periodicity of the week number of every year and the consecutiveness of the last week of the year with the first week of the following year.

On the other hand, we have incorporated new variables coming from other datasets:

- Boots, Sandals, Goretex, Slippers: based on Google Trends searches (more details in Section 2.2).

- Holidays fields: boolean field informing if the Holiday happened that week.

Finally, we have added two new fields: CategoryCluster and ProductCluster, obtained from the clustering study performed in Section 2.5.

2.2 Google Trends Dataset

Google Trends provides access to a largely unfiltered sample of actual search requests made to Google. It works by analyzing a portion of Google searches to compute how many searches have been done for the terms (keywords) entered, relative to the total number of searches done on Google over the same time and geographic area. The idea is to use this information to represent the interest of the population for some specific categories of products in the UK.

In our case we have used the following keywords searches for the corresponding categories:

- Boots: used for categories Womens Casual Boots, Womens Formal Boots and Mens Boots. Added in the final dataset under a feature named *Boots*.
- Sandals: Womens Casual Sandals, Womens Formal Sandals and Mens Sandals. Added in the final dataset under a feature named *Sandals*.
- Slippers: Womens Slippers and Mens Slippers. Added in the final dataset under a feature named *Slippers*.
- Goretex: Womens Goretex and Mens Goretex, under *Goretex* feature

2.3 Currency Rate Dataset

H Dataset includes diverse fields (price, RRP, Markdown, Discount and Cost) with prices using different currencies depending on where the items have been sold. To unify currency, we have used two different datasets with the historical exchange rate from dollars and euros to pounds. Since the different exchange rates are calculated on every weekday and our data is weekly structured, we have taken the Monday value to describe the full week.

2.4 Descriptive analysis

The following sections try to give an idea on what is the structure, values and distribution of the data, their main features and the relations between them.

2.4.1 General measures

In Appendix A (Table A.1) we can find a list of the dataset's features with their relevant measures: mean, maximum and minimum values, standard deviation, number of unique values, etc. But for a more intuitive idea we can look at Figure 2.1, where the histograms are displayed. We can see that sales might vary from 0 to 9281 with a standard deviation of 107.54 and price from £0 to £119.93.

Notice the existence of products with Price, Cost or RRP at zero should not be expected. If we look at the histograms we can see that removing the zero pick from the plots the three variables would follow a distribution similar to a Gaussian distribution. The same is happening with the variable Markdown since it is a calculated field

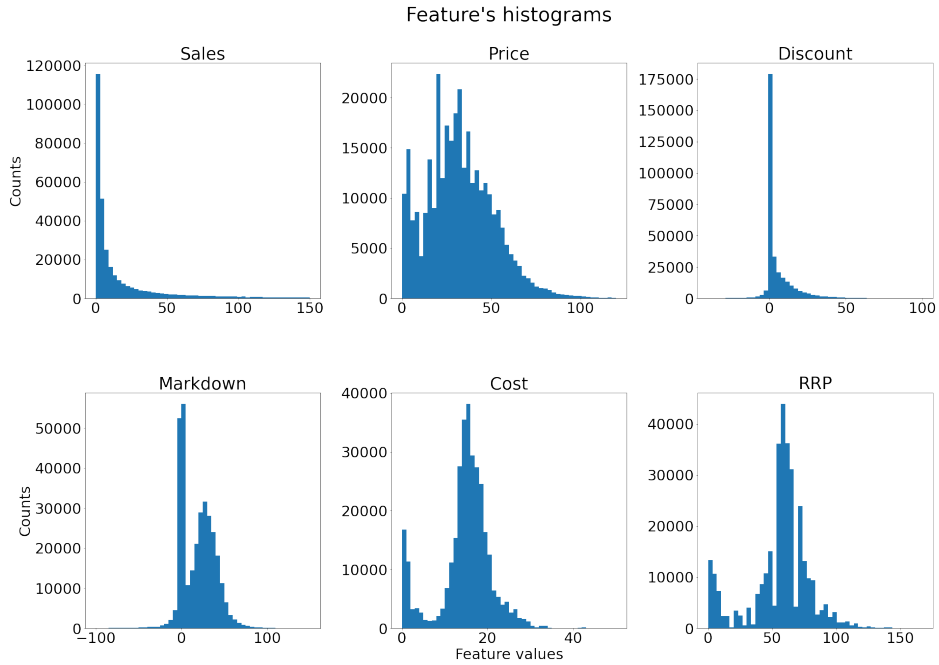

FIGURE 2.1: Plots of the histograms for the relevant features of the H dataset.

following Equation 2.1. For now, we have not removed these values since they contain relevant sales information and most of the values belong to 2021 which would invalidate almost the entire year.

2.4.2 Correlation

In Figure 2.2 we can see the correlation between numerical features. As expected lag variables related to sales (prev_sales, prev_sales2, prev_sales3 and prev_sales4) are correlated between them since the sales of a week are similar to those of previous weeks. We can also see the correlation between price related features where Margin and Markdown variables are inversely correlated. Surprisingly, Discount is positively correlated with Margin (probably due to the fact that if we fix Price and RRP, increasing discounts means decreasing Markdown). This might be due to discounts being independent events that occur when a customer has a special discount and not only related to the sales period.

2.4.3 Sales and price over time

In Figure 2.3 we can see the total amount of sales during the five years represented by week as well as the data aggregation of the sales per month for each of the years. In both plots we can clearly see that the sales during pandemic years (2020 and 2021) have decreased. On the contrary, the prices have had a tendency to increase as shown in Appendix A (Figure A.1).

When studying the sales by category we can observe that some of the categories have a clear seasonality, meaning that the sales tops and valleys happen in the same weeks for different years. In Figure 2.4.a and 2.4.b we can see some examples of this behavior. As expected, Boots are mostly bought during AW season (Autumn and

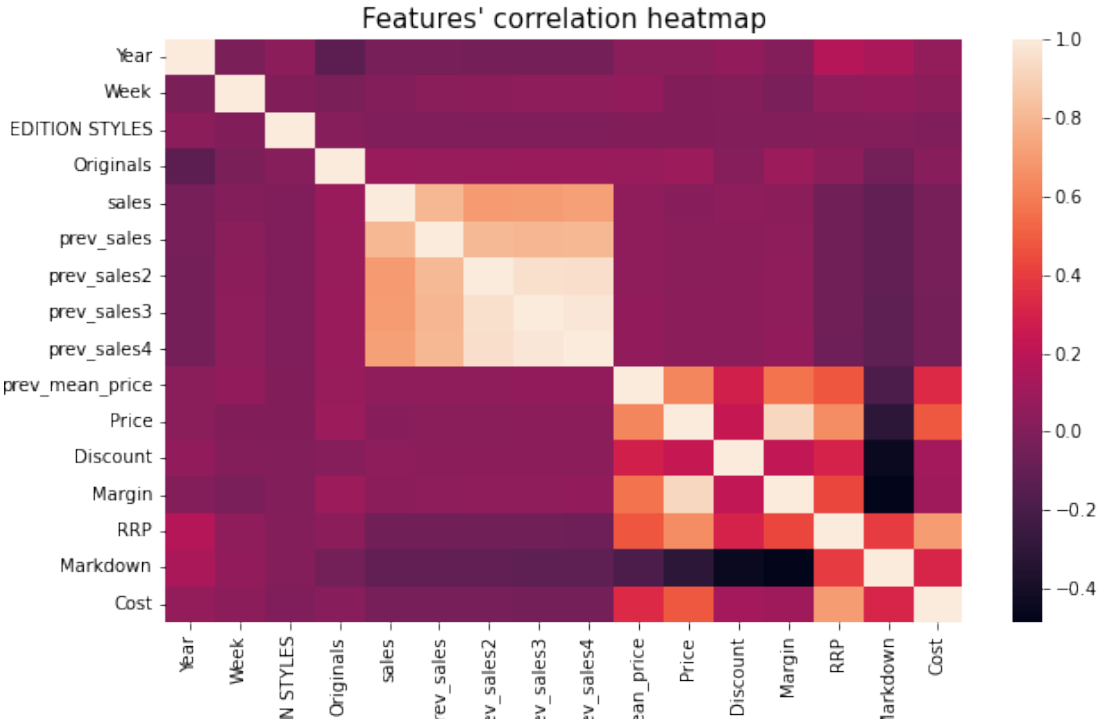

FIGURE 2.2: Heatmap with correlation between numerical features of the H dataset.

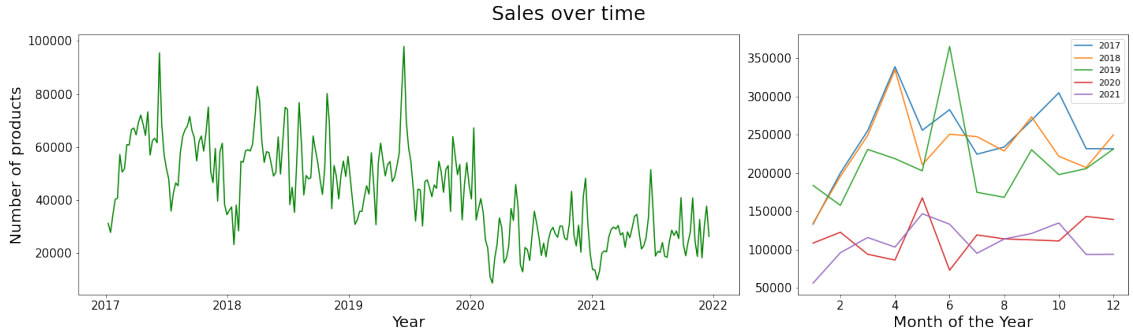

FIGURE 2.3: Weekly sales over the last 5 years (left) and monthly sales for each of the years (right)

Winter) and Sandals on SS season (Spring and Summer). Similar figures for the rest of relevant categories can be found in Appendix A (Figure A.2).

In Figure 2.4.c we have grouped the channels by country and the most relevant country is the UK. Sales are also remarkably different depending on the channel. Within the UK we can observe that around 2020 there was a restructuring of the channels probably due to the pandemic. We can see how Retail - Concessions, Retail - Full Price and Retail - Outlet dramatically dropped (Figure 2.4.d). After that only Full Price modality seems to recover some of the sales. Starting in 2021 new channels appear which group previous channels: UK Offline (contains Concessions and Outlets), UK Online (contains UK Direct channel from 2021 onwards) and Retail (contains Full Price channel).

Finally, if we take a look at the relationship between sales and prices (Figure 2.5) we can see a negative correlation, higher prices have lower demand and the other way around. However, deeper studies will be performed in order to better understand if the relation between them follows the theoretical double logarithm demand model.

2.4.4 Relevant products and Categories

The H dataset contains 1231 different products and 22 categories. The top 5 best-sellers is formed by: Fine Mist Protector Spray, Renovating Cream, Whisper, Shake and Leanne, whereas the most expensive products are: Marlowe GTX, Peak II GTX, Bella, Hydro GTX and Peak GTX (where the mean of the price over time has been used). In this line the categories with more products are: Womens Casual Shoes and Acc Handbags and Purses, and the most purchased categories are: Womens Casual Shoes and Shoe Care.

2.5 Time Series Clustering

As we have seen there are categories with strong seasonality, some of them could even be organised under the same subgroup. If so, we could use models tailored to products or categories with similar time series, and have additional informative features. For that we can use clustering which is a type of unsupervised learning technique that deals with the partitioning of the data into subsets according to a defined distance measure (Madhulatha, 2012). In our case we have used the K-means algorithm (Jain, 2010) for time series[5] which sets a series of centroids, and those time series closer to each centroid belong to that particular cluster. As a result, we are able to set the number of clusters we want and the centroid is the average of all the time series belonging to the cluster.

As a distance measure, we have used Dynamic Time Warping Matching (DTW), a similarity measure which minimizes the effects of shifting and distortion in time by allowing elastic transformation of time series in order to detect similar shapes with different phases (Senin, 2009). In our case, different sales' picks might differ in weeks but still be relevant for the comparison between products or categories.

Clustering has been performed at category and product level. The goal is to find how well are the categories representing the types of products as well as redefining the categories for the modelling, if necessary.

[5] TimeSeriesKMeans from the tslearn library has been used for creating the clusters and the dataset has been aggregated to a record per day.

FIGURE 2.4: Sales over time for a) boots' categories, b) sandals, c) country and d) different UK channels.

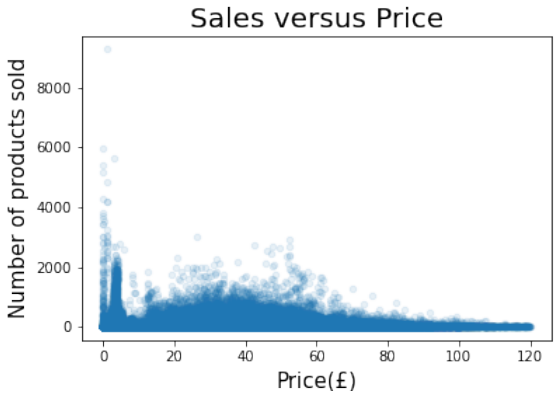

FIGURE 2.5: Sales versus Price from 2017 to 2021.

For the clustering at category level we have created 7 clusters following the Elbow Method (Syakur et al., 2018). All the clusters with their centroids are displayed on Figure 2.6, you can find the correspondence between Cluster and Category as well as the distortion plot for the Elbow method in Appendix A (Table A.2 and Figure A.3). As a relevant output we can see that one of the biggest cluster is composed by categories for men and accessories, while the second one contains Womens Active Shoes, Goretex, Formal Boots, Slippers, Deck Shoes, Formal Shoes, smart Casual Shoes and Mens Casual Shoes. Surprisingly, Womens Formal Boots is closer to Mens Casual Shoes (both belong to Cluster 4) than to Womens Casual Boots (Cluster 5) since they do not share the same cluster.

The idea behind the clusters at product level was to see how good representatives are of the categories compared to the time series of the products that belong to them, as well as see if we can redefine a new set of categories based on the outcomes. We have configured 10 clusters using the Elbow Method (more details in Appendix A, Figure A.4 and A.5). Unfortunately, most of the clusters only had few products on it and 3 clusters stored the majority of the products (89% of the products) which is not ideal if we want to use these clusters as a new categories.

As a summary, we will include the cluster information (product and category level) into our data but only the category cluster data will be used for designing the submodels.

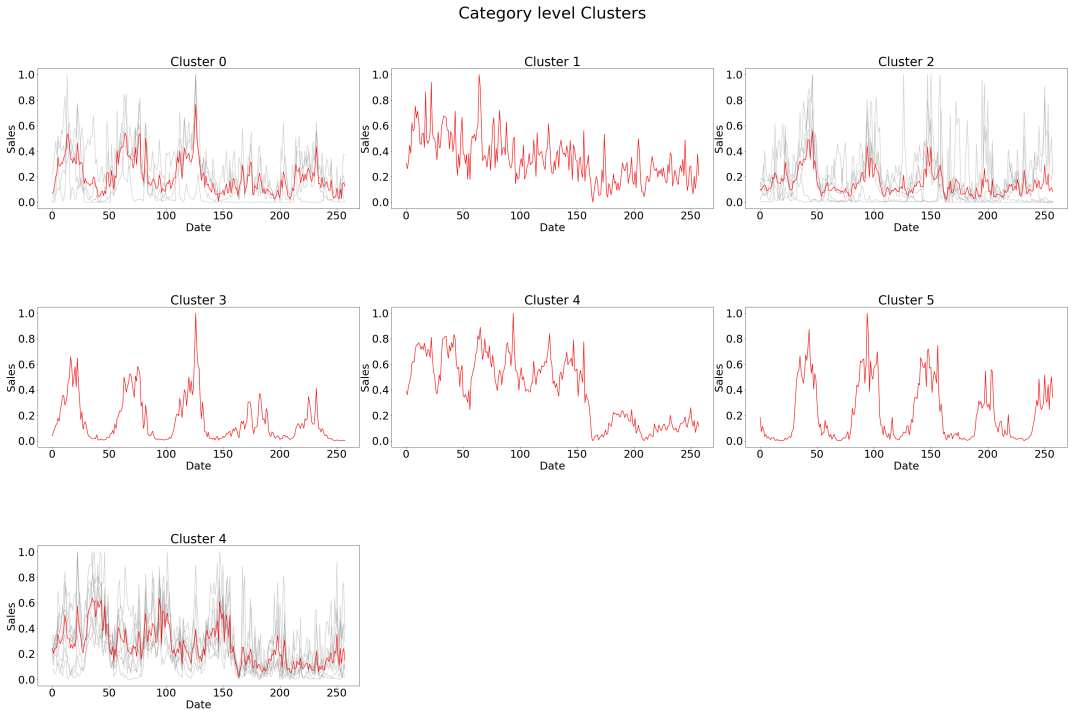

FIGURE 2.6: Plots of the different category clusters, with the category time series (grey) and the corresponding centroids (red). More details can be found on the Appendix A.

Chapter 3

Hypothesis-driven approach

So far we have explained what problem we want to solve and which kind of data we have available. In the following sections we will describe the different approaches and variations we have applied in order to model the data using as a guide the theoretical double-logarithmic demand model equation.

3.1 Model

Our goal is to be able to model the real data using the relation between price and demand. Subsequently the proposed hypothesis space is based on models following the demand function of Equation 1.1. After some transformations we can represent it by:

$$\log D = \log\left(\alpha P^{\beta}\right) \tag{3.1}$$

$$\log D = \beta \log\left(\alpha' P\right) = \beta(\log P + \log \alpha') = \beta(\log P + \alpha'') \tag{3.2}$$

$$D = e^{\beta(\log P + \alpha'')} \tag{3.3}$$

where D is the demand, P the price of the product, α',α'' and β are constants. Equation 3.3 helps us control the demand forcing it to have only positive values.

The loss function to minimise will be an L2 norm fitting of the demand:

$$f_1(w) = \sqrt{\sum_{j=1}^{n} |D_j - \hat{D}_j(w, x_i)|^2} \tag{3.4}$$

where D_j is the demand for the jth data point and $\hat{D}_j$ is their predicted value that depends on: the weights we want to learn w and the features used x_i. This loss function will evolve during the experiments. The learning algorithm used will be gradient descent with adaptive stepsize which is one of the most popular unconstrained optimization methods for N-dimensional problems. It has an iterative structure and as the rest of the general gradient methods follows:

$$w_{k+1} = w_k - \alpha_k \nabla f(w_k) \tag{3.5}$$

where f is the loss function we want to minimize and the stepsize α_k has a successive stepsize reduction of 0.7 if the loss obtained on the k+1 iteration is higher than the previous one (starting value is 0.0001).

For calculating the gradient we have used Autograd (*Autograd Mechanics* n.d.), an automatic differentiation tool that can handle large subsets of Python features and allows us to easily modify our model without the need of obtaining analytically the gradient.

Other methods have been tested, for instance coordinate descent, but dismissed due to the slow convergence.

3.2 Data preparation

The experiments will be performed with a subset of relevant products (12 items) for the most important channel: Retail - Full Price. We have fit a model per product. We have tried also per cluster and per category, but we have decided to work at product level due to its better performance. The interruption of this channel in 2020 is not an obstacle since we are only computing data until the end of 2019. This decision has been made taking into account the elevated number of prices equals to zero within 2021. These data errors might not be that important for a simple demand prediction but they are relevant if we want to focus on the specific relation between demand and price. Our models will be trained with data from 2017 and 2018 and tested with 2019 data since there is a clear change in customers behaviour in 2020 due to the pandemics that our data is not capable of explaining.

3.3 Basic Model

First model tested shows the basic relation between demand and price without specifying any additional information:

$$D = e^{w_0 - w_1 \log P} \tag{3.6}$$

where w_0 and w_1 are the weights we need to learn. In Figure 3.1 we are comparing different plots of the Equation 3.6 while changing the weights. w_0 controls the maximum demand for the minimum price without changing the form of the exponential, hence increasing this parameter we increase the demand for a product at the same price. w_1 controls the strength of the decay in demand as price increases, increasing this parameter would decrease the demand for a product at the same price. For negative values of w_1 we can even change the slope of the tangent since we are converting our function into a positive exponential.

To assure the model is capable of learning we will first test it with synthetic data, specifically generated following Equation 3.6 plus a variable representing the noise. As shown in Figure 3.2 (first row, first column) the model is capable of capturing the tendency of the synthetic data and overcoming the noise.

3.3.1 Results

In Figure 3.2 (first row, second and third columns) we can see the real values of the demand for one specific product and the corresponding sales results using the basic model. As expected the model is not capable of describing the demand only based on the price. During the descriptive analysis we already observed that the relation between price and demand did not follow any particular pattern.

Some of the products are not converging to a viable solution, instead the parameters w_0 and w_1 overgrown increasing the exponential until we get an extremely large value unable to be calculated further.

Since our data has a strong dependence on seasonality we want to test if by removing this seasonality we get the price variations better described. For that we will use a seasonal decomposition.

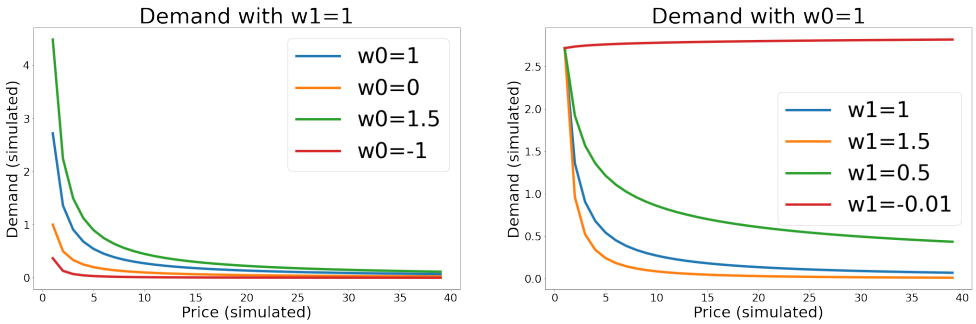

FIGURE 3.1: Plots of the Equation 3.6 while changing the weights w_0 and w_1 remaining the other constant.

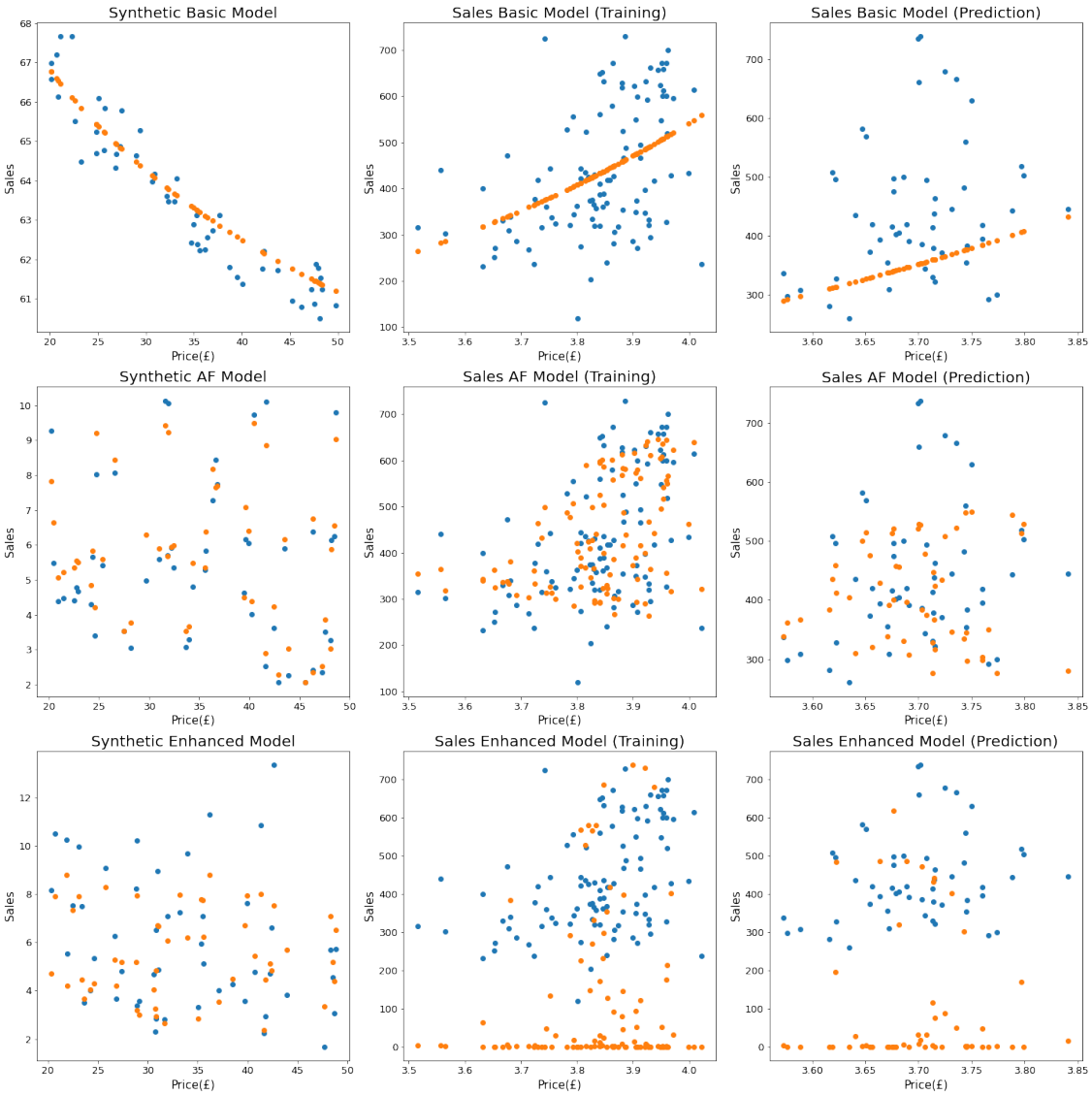

FIGURE 3.2: Synthetic, training and test experiments for the basic model (first row), additional Features model(second row), enhanced learning method(third row)

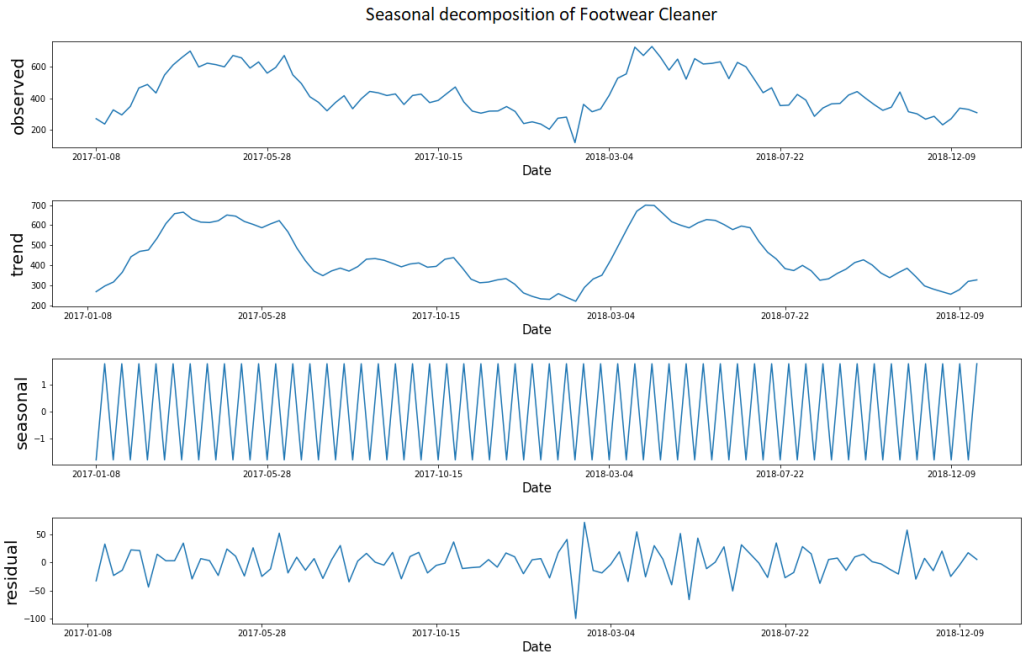

FIGURE 3.3: Time series decomposition of the product Footwear Cleaner: Real data(first plot), trend(second plot), seasonal(third plot) and residual(last plot)

3.3.2 Seasonal decomposition

Our sales data is strongly correlated with factors such as week of the year or seasonality. We have tried to estimate if in absence of the temporal variables our demand variations are related to the price following Equation 3.6. We first need to decompose our time series and identify which components are dynamic and which stationary. For that we use classical additive decomposition (Hyndman, 2018) using moving averages [1] that decomposes our time series in:

- Trend: The increasing or decreasing value in the series.
- Seasonality: The repeating short-term cycle in the series.
- Residual: unexplained variations in the series.

In Figure 3.3 we can observe the decomposition for the product Footwear Cleaner that will be used to illustrate the results in the following experiments. From these 3 time series we can see that seasonality is the one that keeps repeated over time. Hence, if we want to predict future sales we only need to model the trend and the residual and then add the seasonality to the results (we consider our time series additive).

The results obtained modelling the residual are in Appendix B (Figure B.1), but they do not vary from the ones obtained with the Basic Model applied to observed

[1] We have used SeasonalDecompose an statsmodels library that decomposes the series into: trend, seasonality and residual. https://www.statsmodels.org/dev/generated/statsmodels.tsa.seasonal.seasonal_decompose.html

TABLE 3.1: Metrics for all the studied models for the product Footwear Cleaner

Model	RSME	MAE	MAPE
Basic	15916.77	102.94	0.27
Basic Test	20620.33	103.73	0.20
FAM	6285.056	63.66	0.18
FAM Test	9806.64	84.64	0.20
Enhanced	171675.01	370.89	0.84
Enhanced Test	160840.35	362.41	0.82

data. This is due to the fact that the seasonal component is too small, in comparison with the other two, to make a difference and the data points used end up being practically the same.

Since this study has not brought more clarity on the relation between demand and price we should allow the weight to be defined by other parameters in our data, meaning we need additional features to describe its variations.

3.4 Feature Addition Model (FAM)

The idea is to use additional variables from our data to try to explain the results but keeping the same model type and loss function. In this case we follow the equation:

$$D = e^{w_0 - \beta \log \tilde{P}} \tag{3.7}$$

with:

$$\tilde{P} = w_4 x_3 + w_5 x_4 + w_6 P \tag{3.8}$$

$$\beta = w_1 x_1 + w_2 x_2 + w_3 x_3 \tag{3.9}$$

where w_i are the parameters we want to learn and x_i are the additional features we will use. We can interpret $\tilde{P}$ as the perceived price, meaning the actual price that triggers the purchase. This variable will be fed with features related to the price and tries to update the real price with additional information about discounts or previous prices. β represents the disposition of the customer to buy and it is influenced by discounts and temporal variables.

For the experiments we have tried several combinations of features, the one with best results has been: Price, Discount and previous week price (prev_mean_price) for calculating $\tilde{P}$; and temporal variables: Week_sin and Week_cos and Discount for β.

When performing the first check to the model with the synthetic data shown in Figure 3.2 (second row, first column), the results obtained are quite sensitive to the data we create, to the noise, as well as the initialization we choose. As a result, on occasions we arrive into a local minima where the modelled demand prediction is close to 0 for all of the records. Since the data is something we will not change for our dataset we should as well test different initialization in the future.

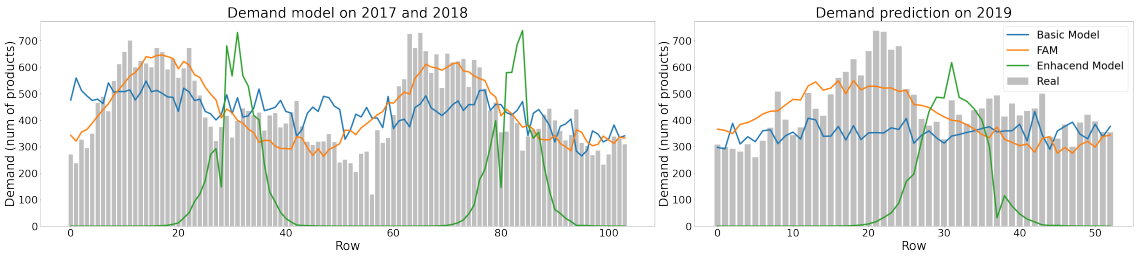

FIGURE 3.4: Real sales in grey modeled by the different models (left) and their predictions (right). Basic (blue), FAM(orange) and Enhanced (green).

3.4.1 Results

The results obtained for the Footwear Cleaner product are in Figure 3.2 (second row, second and third columns). On one hand we have drastically improved the fitting for the training, since the new features allow us to cover more data points. On the other hand, the prediction results have improved from the basic model but still showing some difficulties to reach top sales. In table 3.1 we can see how the metrics have improved accordingly. Unfortunately, most of the products are not converging to a viable solution, on occasions the model overly increases the parameters becoming an exponential of an overgrown value not able to be calculated, or arrives at the local minima where all the sales predicted are 0. However, we have identified that products belonging to the Shoe Care category are more likely to give some viable result.

In Figure 3.4 we can see the sales obtained for each of the rows, notice the axis of abscissas is not the date but the rows ordered by date. This means two consecutive points might belong to the product purchased at the same week at different prices. We can see that FAM starts fitting the real sales for the modelling but shows some difficulties on predicting the increment at the second half of 2019 (similar behaviour to the second half of 2017 during modelling). It also underestimates the pike sales at the middle of 2019 and overestimates the start of the year. Looks like our model fit has a strong dependency on the temporal variables since the shape of the results of the forecasting are really similar to the training part. In Appendix B (Table B.1) it can be found the metrics for the rest of the products of the study.

We should still work to fine tune our model to try to achieve better results and try to guide our model for obtaining less overfitted predictions.

3.5 Enhanced learning method

Our enhanced experiment will be based on the same equation as in the FAM but changing the loss function while keeping the same optimization method.

3.5.1 Loss function

The idea is to guide our model through a loss that penalises the parameters that lead the demand to increase if the price increases and the other way around. We are trying to create a more general model less prone to overfit and achieve a better representation of the unseen data from 2019. For that we take the variables P, $\tilde{P}$ and D, $\hat{D}$ and, considering the real data as a starting point, we define the price and

demand delta as:

$$\Delta P_j = P_j - \tilde{P}_j \tag{3.10}$$

$$\Delta D_j = D_j - \hat{D}_j \tag{3.11}$$

Then, when incorporating this knowledge in our new loss function we have two different definitions depending on the penalty that we want to introduce:

if $\Delta P > 0$ and $\Delta D > 0$:

$$f_2(w) = 2f_1(w) + \sqrt{\sum_{j=1}^{n} |(\tilde{P}_j - P_j) + (\hat{D}_j - D_j)|^2} \tag{3.12}$$

if $\Delta P < 0$ and $\Delta D < 0$:

$$f_2(w) = 2f_1(w) + \sqrt{\sum_{j=1}^{n} |(P_j - \tilde{P}_j) + (D_j - \hat{D}_j)|^2} \tag{3.13}$$

where $f_2(w)$ is our new loss function and $f_1(w)$ the original loss function from Equation 3.4. This means that if the perceived price is higher than the observed price and the predicted demand is higher than the real demand our loss will penalise it. The same happens if the perceived price is lower than the real price and the predicted demand is lower than the observed demand. We have given a weight of 2 to the demand fitting original loss (f_1), since we want it to have more importance than the rest of the new loss.

During the synthetic check shown in Figure 3.2 (third row, first column) we can see the model successfully learns even though the computing time is much higher and the results are less fitted than before.

3.5.2 Results

Results are displayed in Figure 3.4 (third row, second and third columns) and Figure 3.2, where we can see that modelling results have worsen for the training and the prediction. The results are far from fitting the real data, showing a narrowed pike each year that is some months delayed from the observed pike. Sales are hugely underestimated for the range of points out of the picks. Looks like our method is not capable of guiding the modelling of the data we have. Table 3.1 show how all the error metrics have increased as well.

3.6 Demand matrix and optimal price

The demand matrix is an N-dimension matrix storing the information of the different demands obtained by combining feature values. In our case the demand matrix will be a vector with the different demands for different prices. Depending on the problem additional dimensions as week or channel can be added.

In order to calculate the matrix we first obtain the historic mean of the price as well as the mean of the unitary cost. From here we create a range of prices higher and lower than the mean price. We then apply the model to our data with the parameters learned in our Feature Addition Model (since it is the one that performed the best).

The idea is to optimize the price in order to get the maximum benefit. We define the benefit as:

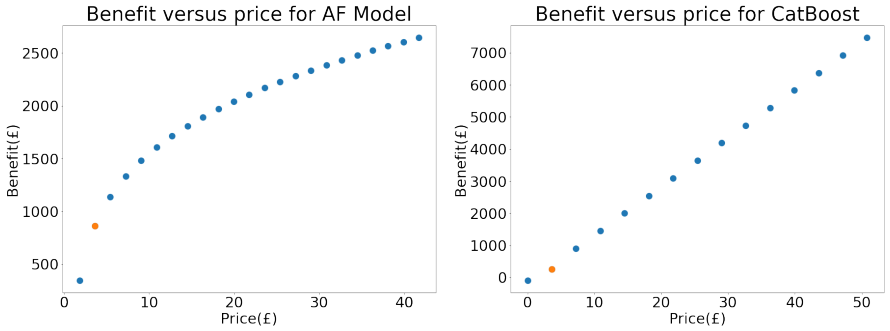

FIGURE 3.5: Benefit versus price based on demand predicted by FAM. Mean historic price marked in orange.

$$B = D(P - c_p) - c_f \tag{3.14}$$

where B is the benefit, c_p the cost of production (corresponds to the unitary cost we have in the dataset) and c_f is the fixed cost that we take as 0 since we do not have this information available.

3.6.1 Results

In Figure 3.5 we can find the relation between the benefit and the price based on the predicted demands of the FAM. The orange dot indicates the historic mean price, and we can observe that increasing the price always increases the benefit. This result is not realistic, since it means that we should increase our price more than 1300% of the mean price. A product at that price will probably not be bought by almost anyone. This also shows some of the weakness of our model and that tends to overestimate sales for elevated prices. The parameter able to modify the decay of the demand as the price increases is β, the disposition of the consumer to buy. A higher β would diminish the demand and as a result our benefit would also be lower for high prices.

3.7 Discussion

Our model has shown the capability of modelling and predicting certain products with good results. We have also shown that adding more features in the definition of our features has improved our results and make the model capable of covering the different demands for the different prices.

Even though the fitting of the data has been satisfactory, the prediction could be improved. We encountered predictions too conservative where the pike of the sales are not achieved.

3.7.1 Model limitations

Our model has shown some limitations that prevent us from using it in real life cases. The most relevant one is that our model is not applicable to all of the products. We have experienced that mainly products belonging to the Shoe Care category are responding to our model. One of the reasons might be Shoe Care is the only category

with one unique value for the Seasonality field. The rest of the categories combine the values of the Seasonality feature during the different weeks (in the same week you can find purchases of a product with the Seasonality SS and AW). This adds a variability on the prices that is not reflected in the model since we are not using the categorical values of the Seasonality feature.

Our model is based on the relationship between demand and price but our data actually comes from imperfect observation of the true demand due to the demand censoring effect, where the actual demand exceeds the available inventory. Out of stock products are tracked as lack of demand due to the limited amount of sales. At the same time, when facing a stock-out in the primary target product, customers may turn to purchase substitutes, which may increase the sales of substitute products and result in an overestimate of the demand for them (Fildes, Ma, and Kolassa, 2019).

We do not have information regarding the advertisement or promotional campaigns that would affect our β, changing the disposition of the customer to purchase.

Since our model is at the product level we are not capable of dealing with inter-category effects. Our product data is ignoring the behaviour of the rest of the products since we do not have other products' information on it.

Due to the large computing time our learning method spends in finding a minima for a single product and channel we have tried to simplify as much as possible our demand Equation 3.7 and restrict the number of variables used. This is also a limitation since we are limiting ourselves to use only some of the data available.

We can not exclude the existence of unobserved confounders. A confounder is a variable that influences both the dependent variable (D) and independent variable (P), causing a spurious association (variables are associated but not causally related). For instance, shoe quality could be a confounder. Higher quality shoes might have a higher demand than low end shoes, as well as higher price due to the materials used. Without this variable our algorithm might assume higher prices imply higher demand, which is not true.

Finally, our model is based on trained parameters per product with historical data making it difficult to deal with the cold-start problem, therefore to predict new products without previous data.

Chapter 4

Data-driven approach

As discussed in the previous section our custom model is not applicable to all the products. We need to look for an alternative as well as compare the FAM with another method for the evaluated products in the previous section. The chosen machine learning algorithm for performing these checks is CatBoost. Gradient boosting models have achieved good results in a multitude of use cases, they have high scalability, they allow non-linear relations between variables and, specifically CatBoost, has great advantages for dealing with categorical data, using Ordered Target Statistic technique (Prokhorenkova L, 2018).

4.1 CatBoost

CatBoost [1] is an open source library algorithm that uses gradient boosting on decision trees. It belongs to the family of gradient boosting algorithms which combine a weak prediction model (decision trees in our case) iterated into a strong learner (gradient descent). In addition, it uses ensemble techniques where models are added sequentially to the ensemble, correcting the performance of the prior ones.

CatBoost modifies the computation of gradients to avoid the prediction shift in order to improve the accuracy of the model (Bentéjac, 2021). This shift occurs because during training gradient boosting is using the same instances for the estimation of both the gradients and the models that minimize those gradients. To overcome it, CatBoost estimates the gradients using a sequence of base models that do not include that instance in their training set (Prokhorenkova L, 2018).

We have used CatBoostRegressor since we have a regression problem. We have built a model for each Category Cluster and after gridsearching on the parameters we have taken: maximum number of iterations of 60, with a tree depth of 11, a learning rate of 0.1 and RSME as a loss function. For choosing the parameters we have considered the global metrics of the results.

The features used in the Catboost algorithm are: Description, Seasonality, Year, Week_sin,Week_cos, Channel, WSSI Category, Originals, Source, gender, style, Price, Discount, RRP, Markdown, prev_sales2, prev_sales3, prev_sales4, Cost and Product-Cluster.

4.1.1 Custom Model and CatBoost algorithm results

In Table 4.1 and Figure 4.1 we can see the metrics and results for Footwear Cleaner with the FAM and the CatBoost algorithm. For the training set, the results for Catboost have a better fit on the real data and all the metrics have better scores. For the forecasting, CatBoost has a better fit as well, being able to get the tendency to

[1] https://catboost.ai/

TABLE 4.1: Metrics of the demand modelling and prediction for Footwear Cleaner (Retail - Full Price channel).

Product	Experiment	Method	RSME	MAE	MAPE
Footwear Cleaner	Training	CatBoost	3142.67	43.47	0.12
		FAM	6285.056	63.66	0.18
	Test	CatBoost	9077.83	74.45	0.16
		FAM	9806.64	84.64	0.20

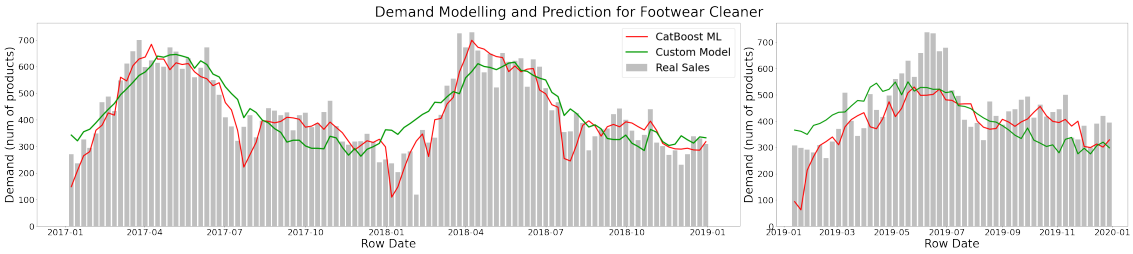

FIGURE 4.1: Demand modelling and prediction for Footwear Cleaner (Retail channel) using Catboost (red) and the FAM (green). Real data is represented in grey bars.

increase or decrease correctly, even if the maximum of sales is not achieved. The metrics are also better for CatBoost in this case.

4.1.2 Discussion

CatBoost is overperforming our FAM for those products that we have been able to evaluate with the Custom Model. Specially on the prediction part, Catboost is able to describe the tendencies of the sales while FAM is only capable of middling forecasting the first sales pick. CatBoost is able to manage higher dimensional problems since it is able to include a lot of features (some categorical) that help to shape our sales forecasting.

Let's see how well CatBoost performs for the rest of the products and compare it to a linear regression model for completeness.

4.2 Overall CatBoost Results

CatBoost algorithm has been trained from 2017 to 2019 for comparing the results with FAM but also from 2017 to 2020 to be able to see how effective the method is under change of social behaviour. Additionally, in CatBoost the relation between sales and rest of the features is not focused on the price, hence odd values are not as relevant as for FAM.

After training our machine learning algorithm based on category clusters from 2017 to 2020 we have obtained a MAE of 17.92 and a RMSE of 2371.75 while in the prediction for 2021 the values are 21.12 and 3727.65, respectively. For reference a model that predicts the mean of the sales would score a RMSE of 4881.37 and a MAE of 30.80 for the forecasting of 2021 and RMSE of 12785.04 and MAE of 46.20 from 2017 to 2020.

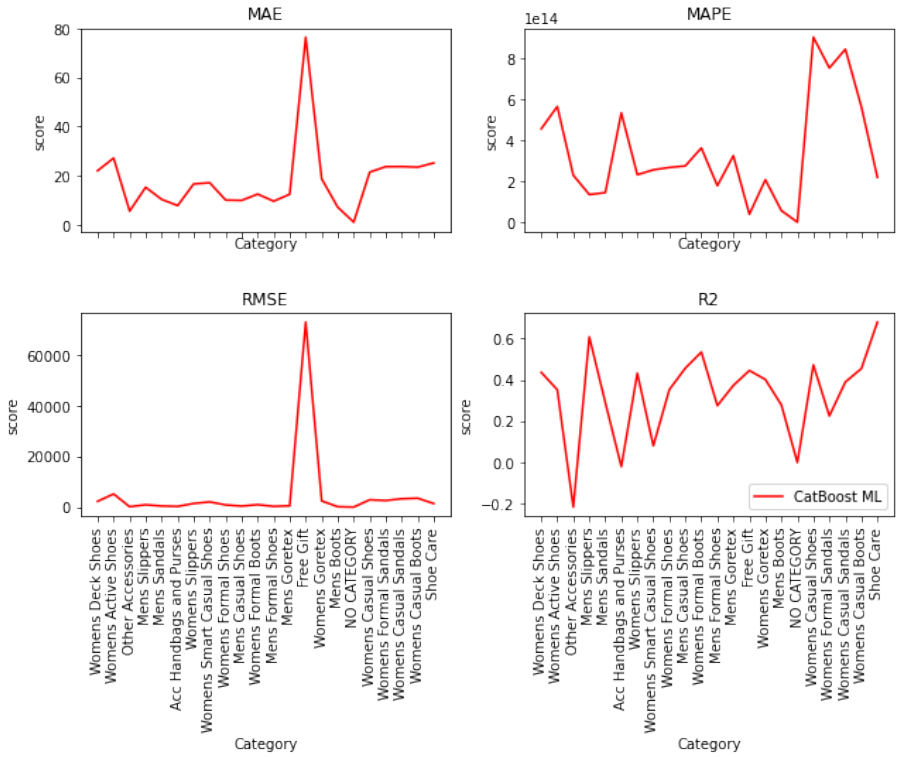

FIGURE 4.2: Metrics for the CatBoost predictions plotted by category.

In order to understand how good we are predicting for each category we have plotted the different metrics in Figure 4.2. The category that shows more difficulties in being predicted is Free Gift. From a business perspective this category is quite irrelevant since it is not inventory dependent and the Free Gift sales are later translated into other categories' sales. It is also one of the categories with less data.

4.3 CatBoost and linear Regression tool

In order to evaluate the CatBoost algorithm we will compare it with a linear regression tool designed and used for demand prediction in retail called ROVA. In this case we are not limiting our dataset to year 2017, 2018 and 2019 but taking the full 5 years and using from 2017 to 2020 for training and the 2021 data for forecasting.

4.3.1 Linear Regression

For the linear regression we have used ROVA tool, a platform specifically designed for sales forecast and currently used in several retail companies. This allows us to compare the results of CatBoost with a real life case tool. ROVA is based on Python Stats Models and allows us to use linear regression with random effects for a limited number of variables. It is also capable of visually displaying the results and modifying the model close to real time.

We have performed two experiments, a model for the overall data and a model for an example category. Since Category specific models performed much better than general ones we will directly display the first ones. The chosen category has been Womens Goretex since it is a category that performs as the average on the

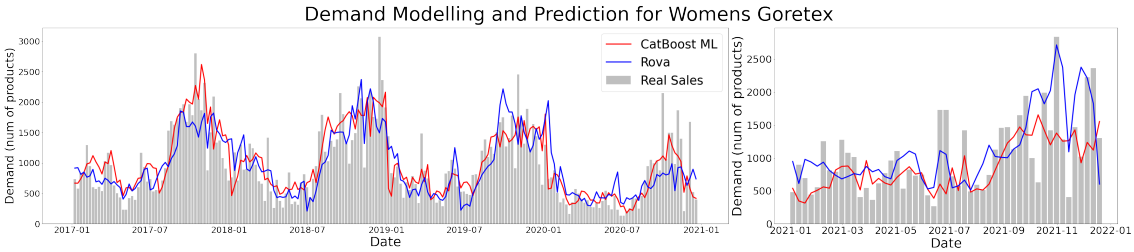

FIGURE 4.3: Modelling and prediction for category Womens Goretex with CatBoost (red) and ROVA (blue).

metrics (see Figure 4.2) in the CatBoost Model and also because it allows us to use the Google Trends field based on the keyword Goretex.

Due to a platform limitation, the data needs to be grouped by date. This means we need to aggregate our dataset in order to have one row per week. Within the aggregation we are taking the mean of the Price, Discount, Cost and Markdown, losing some information. In order to be able to compare it with the CatBoost results we have aggregated CatBoost results as well.

It is also worth mentioning that we have used the Holidays Dataset information in the regression since it helped us to shape the linear model. Random effects were applied to the variables Price and Markdown.

4.3.2 Results

In Figure 4.3 we can see training and forecasting of ROVA and CatBoost results for the category Womens Goretex versus the real sales for each week. Both methods show great results. For the modelling, while Catboost has a more accurate and smooth behaviour, capable of predicting some of the top sales week, ROVA has a more erratic outcome over the model. None of the models accurately describes most of the top sales weeks. On prediction time ROVA's result shows higher discrepancies between the real and the predicted data than CatBoost, specially in the beginning of the year but it is predicting the top sales weeks at the end of the year. CatBoost results are more fitted to the real data but with a tendency to underestimate sales. This is proved as well on the metrics displayed in the Table 4.2 where CatBoost outperforms ROVA in terms of all the metrics.

In Appendix B (Figures B.2, B.3, B.4 and Table B.2) we can find the results for individual products. We have picked 3 different items: for the first product we have data from 2019 to 2021, for the second from 2017 to 2021 and the third one has only been sold during 2021 (used to simulate the cold-start problem).

The predictions for the 3 products are remarkably different. While for the first product, ROVA is overestimating the sales for all 2021, for the second product is actually performing slightly better than CatBoost.

When asking the models to predict a product that they have never seen before, CatBoost is delivering a good performance but not achieving to predict the sale pikes. For ROVA we see the results are not accurate during the first months of 2021 but are able to better approximate the maximum sale weeks.

The overall behaviour of CatBoost is better than the linear regression but we have encountered few exceptions where ROVA is predicting better than the machine learning algorithm.

TABLE 4.2: Metrics for the Category Womens Goretex using CatBoost and ROVA.

Experiment	Method	R2	RSME	MAE	MAPE
Training	CatBoost	0.66	115803.26	262.22	0.34
	ROVA	0.56	149113.16	288.02	0.38
Prediction	CatBoost	0.21	249973.27	342.52	0.33
	ROVA	0.19	256389.63	394.00	0.50

4.3.3 Linear, CatBoost error distribution

In Figure can find the errors on the predictions for the different demands and product prices for CatBoost and ROVA. When comparing the error for both methods: ROVA and CatBoost, we observe how results tend to underestimate high sales and overestimate the values in the valleys. This is an expected result since real data always has some noise and machine learning methods tend to generalise a solution that works for most cases. This explains why extreme data points are usually badly predicted.

For the error in the prediction CatBoost tends to underestimate high sales and ROVA tends to overestimate low ones. This translates to CatBoost predicting more accurately expensive products (with low demand) and ROVA performing better with best sellers.

4.4 Explainability for CatBoost

Explainability is an important property when discussing demand forecasting, specifically if it is done for price optimization purposes. Companies request to be convinced and understand the reasons for the results and predictions before using them for sensitive topics directly related to revenue.

The need for interpretability arises because for certain problems or tasks it is not enough to get the prediction. The model must also explain how it came to the prediction (Molnar, Casalicchio, and Bischl, 2021).

CatBoost has a builtin solution for explaining the predictions, TreeSHAP(Lundberg et al., 2020). It is a variant of SHAP and is a local White-box method that can be used in XGBoost, LightGBM, CatBoost, scikit-learn and pyspark tree models. It can be used for classification of regression problems.

The goal of SHAP is to explain the prediction of an instance by computing the contribution of each feature's value to the prediction.

We need to take into account that with TreeSHAP features that have no influence on the prediction can get a TreeSHAP value different from zero. The non-zero estimate can happen when the feature is correlated with another feature that actually has an influence on the prediction (Molnar, Casalicchio, and Bischl, 2021).

Since we have a model per cluster we have a set of Shap values for each. We are going to focus on the Cluster 0 that contains Womens Goretex category (and others). In Figure 4.5 we can see the Shap Values plotted. The features are on the left vertical axis ranked in descending order according to importance and the Shap value strengths are on the horizontal axis. The horizontal location of the dots shows whether the effect of that value is associated with a higher or lower prediction and

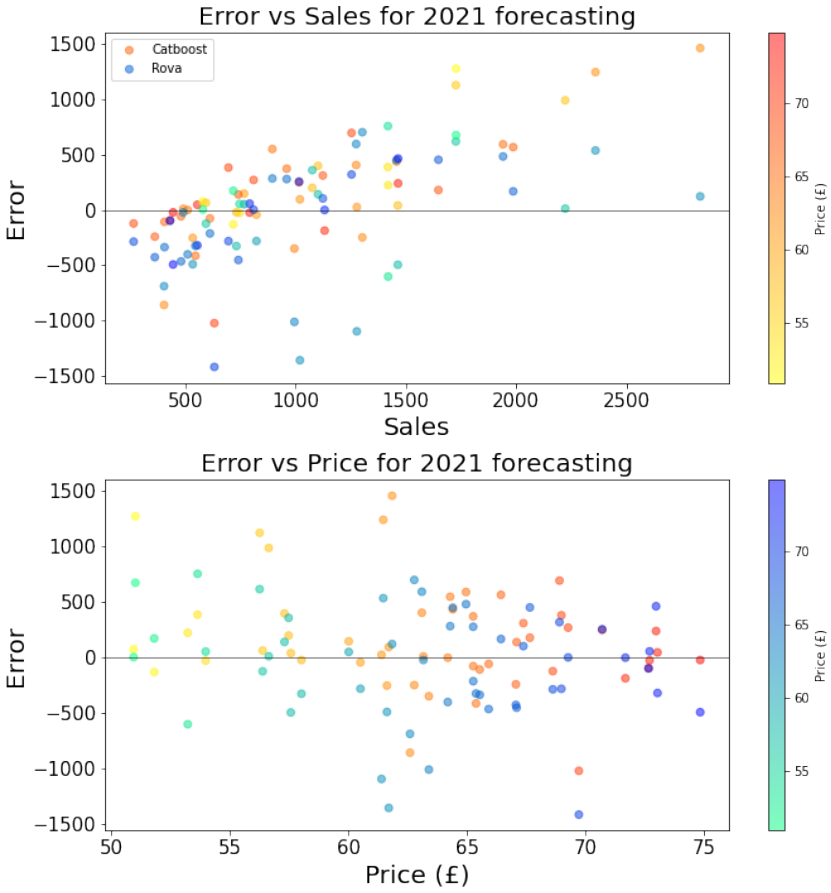

FIGURE 4.4: Error versus sales (top) and error versus price (bottom) for the demand prediction of Womens Goretex of 2021.

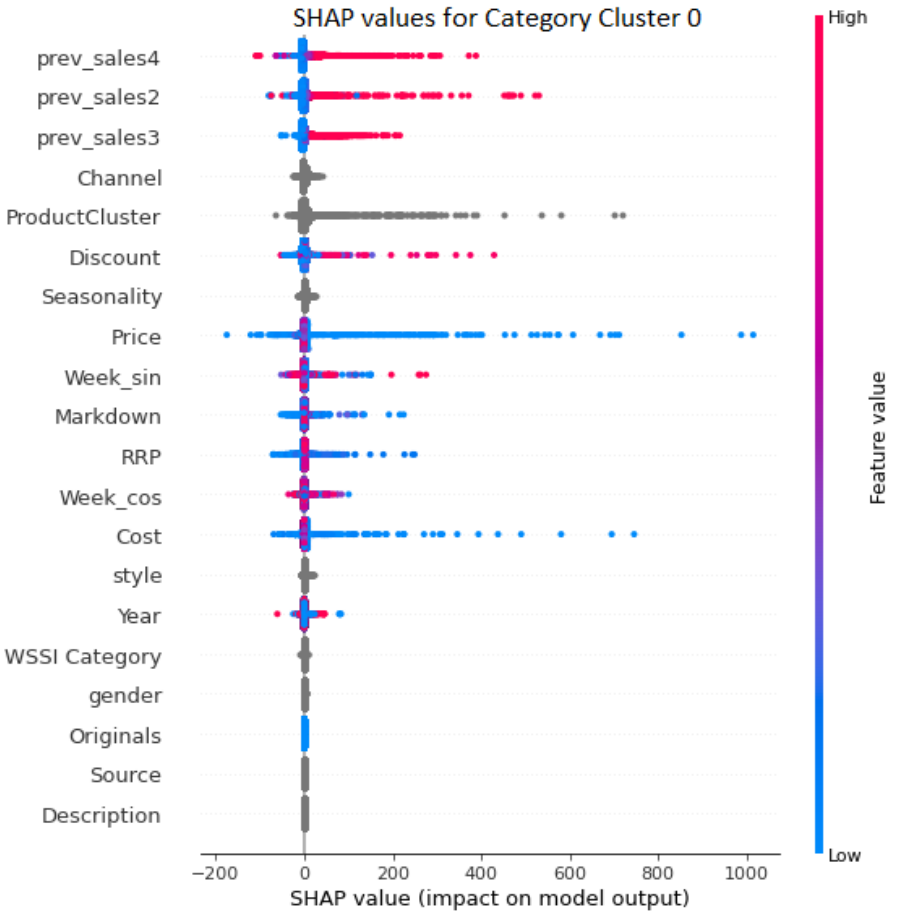

FIGURE 4.5: Shap values for the Category Cluster 0

the color shows whether that variable is high (in red) or low (in blue) for that observation. Variables in grey correspond to categorical features.

We can see the top 3 most relevant features are the sales information that previous weeks provide. High values affect incrementing the outcome value and low diminishing it. Channel is the fourth feature probably due to the difference of sales between channels. We can also see that price is the eighth variable of importance.

In Appendix B (Table B.3 and Figure B.5) we can find the plot on the average importance of the features in the model and the top 10 interactions between variables.

4.5 Demand matrix and optimal price for CatBoost

In Figure 3.5 we can see the optimization function obtained with the CatBoost algorithm for a product example. Even though the best results on demand forecasting come from CatBoost algorithm, the price optimization function plot shows that the benefit increases linearly with the price, which is not realistic since this would mean that our optimal price is infinite. If we take a deeper look we can realise that our demand matrix has an odd behaviour. The price values we are considering go from 0 to 754, however our demand vector predicts the same demand for all of the prices higher than the mean price.

4.6 Discussion

CatBoost has shown to be the best of the three methods for demand forecasting. Their overall predictions are satisfactory on test and training and they keep being good when diminishing granularity. Our FAM and ROVA also show good results but with higher limitations.

CatBoost has shown to be great for expensive products while ROVA is better at predicting the sales for the best sellers.

However, CatBoost has shown as well some limitations when predicting our demand matrix. The reason might be the intrinsic tree structure of the method. For example, when using a tree as a classifier we are setting conditions for deciding if our input is A or B. But when using a tree as a regressor we are somehow defining a range of viable results for our outcomes. If our previous data do not contain similar cases to the one we want to predict, the algorithm will probably not be able to calculate the new result and stick to the closer *known* data point. Other methods with a stronger mathematical base might be better for dealing with this type of counterfactual queries instead of tree structures.

In this sense, CatBoost is good at forecasting because the past is somehow similar to the future and the variation of inputs and results allows the algorithm to give an overall great prediction. It is when we want to ask counterfactual reasoning with unseen behaviour when the algorithm fails.

Chapter 5

Conclusion and future work

In this study, we have applied different techniques to predict the demand of different products in a case study with real data. We took firstly an hypothesis-driven study where different experiments were tested (Basic, FAM and Enhanced learning method) and, secondly, a pure machine learning approach (CatBoost).

Among the different methods and model fits performed, we have observed that the best variation on our Custom Model was FAM. FAM is capable of modelling and predicting the demand but it is not advanced enough for fully predicting the sales' pikes. In addition, it has also shown some limitations since our model is only useful for certain products. The enhancements and guidance we have tried to apply through our new loss function have not improved the forecasting (or the modelling), however the study provides us insights for future work. If the computational time can be optimized and we achieve to incorporate more variables (e.g. field Seasonality), improving β (parameter that controls the disposition of the customer to buy), we could obtain better results.

CatBoost has been proven the best model compared to linear regression and FAM. Predictions overcome those of FAM and it has been shown that CatBoost modelling and prediction of sales have great results. Compared to linear regression, CatBoost predicts more accurately expensive products while ROVA performs better with best sellers. However, we have found a limitation of the method when calculating the demand matrix, since at some price limit our sales predictions stopped changing. Looks like the tree structure of the algorithm is not great for dealing with counterfactual queries regarding unusual prices.

We can conclude that CatBoost is skillful at demand forecasting at product and category level, and it is also good at predicting cold-start products if we know the category they belong to. CatBoost has as well been superior to our FAM and linear Regression models.

However, CatBoost is not correctly performing for price optimization tasks. In this sense we believe that if further improvements and investigations are carried out, FAM would achieve better results than CatBoost on calculating the optimal prize for maximising revenue, which is the ultimate goal of this study. Additionally, a proper optimization algorithm should consider not only the price and unitary costs, but also factors such as advertising costs, fixed costs, inventory and competitors' prices (and stocks).

Our hypothesis was focused on a relation between the demand and the price but, as we have seen, the data we have is not representing the demand but the sales. The demand censoring effect affects this relation and to overcome it we should have at least the stock data. There is also other data that would be interesting to have that affect the perception of the price and the disposition to buy like advertisement or promotional data.

This study also shows the importance of industry knowledge. Companies acquire expertise that is then transmitted to their decisions and final prices. The complexity of collecting all this intangible information in our data will hinder machine learning algorithms to grasp the real relations between demand and price or to find the optimal prices.

5.1 Future Work

Some lines of work that could be developed to further enhance this study are:

- Perform a hierarchical clustering analysis (Nielsen, 2016), which is an unsupervised technique that shows how clusters evolved from one-item cluster to include all the items in one unique cluster. This technique helps us to understand the cluster hierarchy of our time series through a dendrogram. Also considering alternative distance measures could be useful for better redefining our present categories.

- Try to improve our FAM model by including additional variables for example Seasonality or previous sales, and focusing on improving our β (customer's disposition to buy). This would as well help in our demand matrix and our price optimization task by extension. Prior to that we would need to enhance the learning method for diminishing the computational time.

- Extensive feature engineering study for the CatBoost Model, where we transform the variables based on the double-logarithmic demand model. For instance using the logarithm of the price or sales. This might improve the outcome but the demand matrix limitations would persist. Another option is to try other models not tree based to revise how they behave with the retail data, for instance K-Nearest Neighbors Regression or Deep learning (if a local black box method can be applied for dealing with the explainability part).

- Use other techniques for guiding the model. For example: Oversampling methods, where synthetic data is created following the double-logarithmic demand model. In this way, well behaved data would help the model to find the theoretical relation between the sales and the price, without focusing on edge cases where this relation is not followed.

- Exploring Counterfactual reasoning is definitely an interesting point, since the statements we are trying to solve are "if" statements in which the hypothetical condition or antecedent is untrue or unrealized (Pearl, Glymour, and Jewell, 2016). For example: If I had set the price to x, how much would I have sold? These models are based on causal inference which would overcome confounding issues. Nevertheless, most of the experiments performed with this technique are based on binary variables. We would need to adapt our data or use alternative techniques to perform this type of reasoning.

- Price optimization improvements by taking into account stock, advertisement, costs (per channel, fixed and storage), competitors' prices and price aesthetics. Most of these data was not available for us but some competitor's data can be obtained using web scraping techniques. Regarding this point it is also important to take into account that price strategy might affect your own products and that the data needs to be cross evaluated through all the products to maximise the overall revenue.

Appendix A

Dataset information

A.1 Dataset Features

H dataset contains the following features and their description:

- Description: Name of the product, it works as an identifier of the product. We have 1231 unique values on it.
- Seasonality: Seasonality of the sales, we can found the following values: SS:Spring/Summer, AW: Autumn/Winter, OLD, CONT and NO SEASONALITY.
- Month: Month of the sales record, contains Nan for all the rows.
- Year: Year of the sales record, contains data from 2017 to 2021.
- Week No: Record's week of the year. Values go from 1 to 53.
- Channel: Which Channel the sales belongs to. The channels available are: DE Direct, Retail - Concessions, Retail - Full Price, Retail - Outlet, UK Direct, US Direct, Wholesale, Euro Direct, Digital Partnerships, EU Direct, Retail, UK Offline, UK Online, US Offline, US Online.
- WSSI Category: Category of the product. We have the values: Womens Formal Shoes, Womens Active Shoes, Womens Casual Shoes, Womens Slippers, Mens Casual Shoes, Womens Formal Sandals, Womens Casual Sandals, Womens Goretex, Womens Deck Shoes, Acc Handbags and Purses, Mens Formal Shoes, Womens Casual Boots, Womens Formal Boots, Mens Slippers, Womens Smart Casual Shoes, NO CATEGORY, Shoe Care, Other Accessories, Mens Sandals, Mens Goretex, Free Gift, Mens Boots, Mens GoreTex and Womens GoreTex.
- LIMITED EDITION STYLES: Contains information regarding if the product is (or not) a limited edition. Values: NON-LIMITED EDITION, LIMITED EDITION and empty values.
- H Originals: describes if it is an original product of the Brand. Contains values: No, Yes and empty.
- Source: source of the product. Can have the values: MANUFACTURED, SOURCED or empty.
- RRP: Recommended retail price, is the price at which the manufacturer suggests the retailers to sell its product.
- MARKDOWN: reduction of the product price (RRP), based on the inability to sell it at the initial price or original selling price.

TABLE A.1: Metrics of numerical variables of the H Dataset

Metrics	Sales	Price	Discount	RRP	Markdown	Cost
mean	34.05	30.22	4.58	52.55	17.76	13.89
std	107.54	19.49	11.17	26.32	20.06	7.63
min	0.00	0.00	-368.81	0.00	-328.34	0.00
25	2.00	16.67	0.00	45.77	0.00	11.99
50	5.00	29.17	0.33	57.50	16.67	15.13
75	24.00	42.83	5.84	65.83	32.50	17.90
max	9281.00	119.93	345.49	167.37	406.76	1619.86

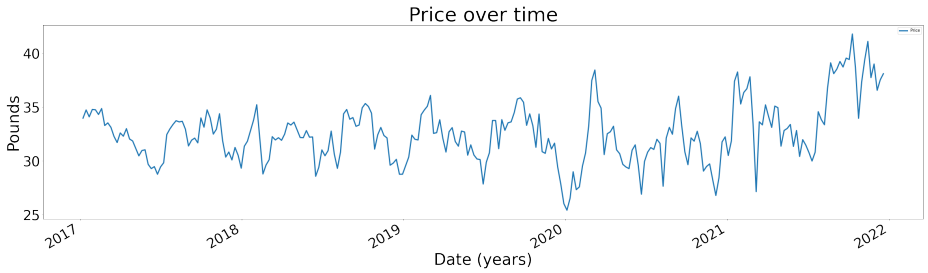

FIGURE A.1: Prices over time

- NET DISCOUNT: Aggregated data with the discounts during the week.
- NET REVENUE: Aggregated data with the Revenue during the week.
- NET VOLUME: Aggregated data with the items sold during the week.
- NET MARGIN: Aggregated data with the margin obtained during the week.
- COST: Aggregated data with the costs during the week.
- Season Report: Report in which the sales are added internally. Can have values: SS17, AW17, SS18, AW18, SS19, AW19, SS20, AW20, SS21 and AW21.

In table A.1 we can see relevant measures(mean, standard deviation, minimum and maximum values) for the numerical features.

A.2 Features over time

In Figure A.1 we can see how the price has evolved and increased during the 5 years.

In Figure A.2 we have displayed the sales for different Categories over time.

A.3 Time Series Clustering

Table A.2 displays the different product categories with the corresponding Category Cluster. And in Figure A.3 and A.4 we have the plot for the Elbow method determining the number of clusters we have chosen for category and product level. Finally, in Figure A.5 we find the different clusters at product level.

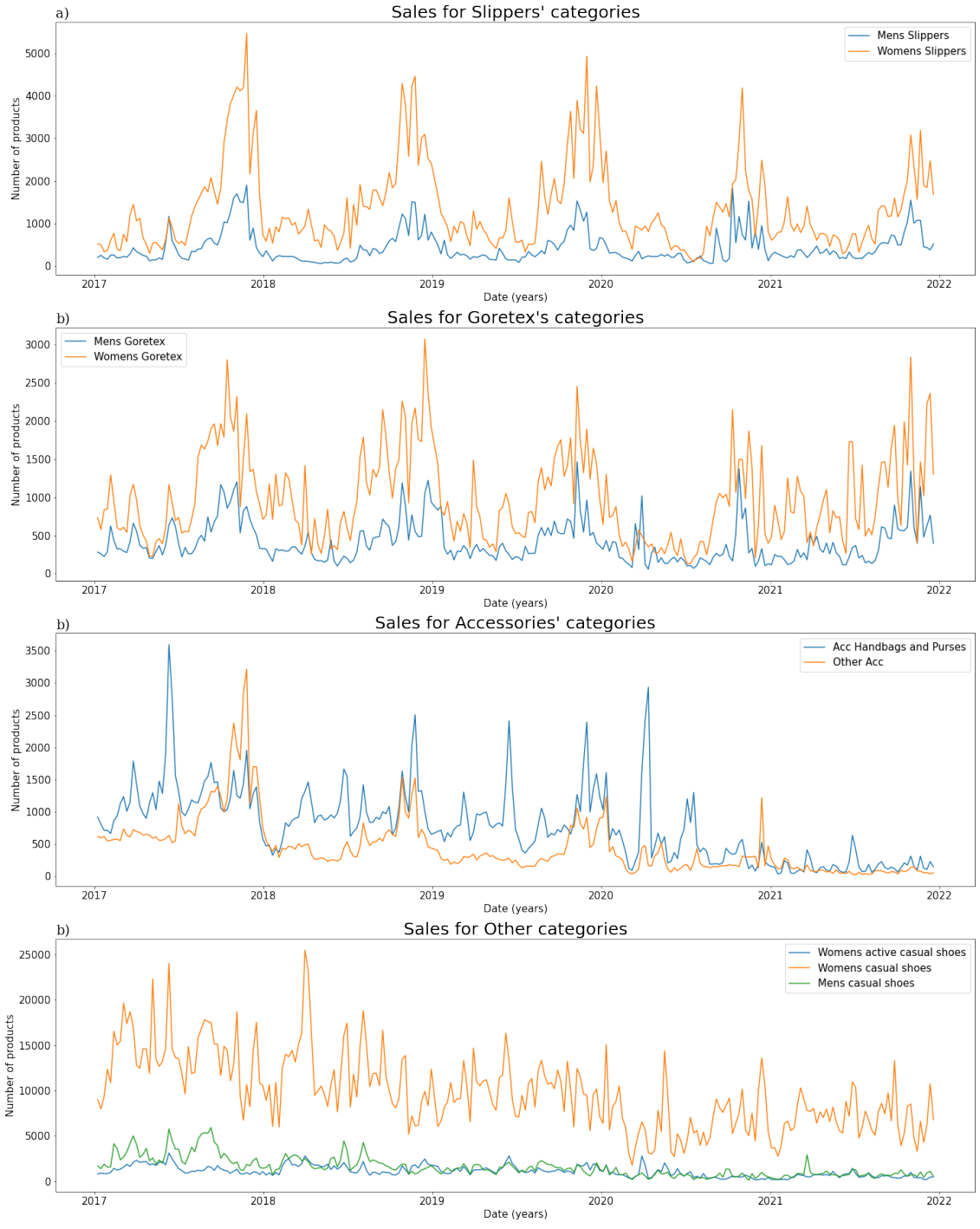

FIGURE A.2: Sales over time for a) Slipper categories, b) Goretex Categories, c) Accessories and d) Casual categories.

TABLE A.2: Correspondence between Cluster and product category

Category	Cluster
NO CATEGORY	Cluster 0
Womens Goretex	Cluster 0
Womens Formal Shoes	Cluster 0
Womens Formal Boots	Cluster 0
Womens Slippers	Cluster 0
Other Accessories	Cluster 0
Mens Slippers	Cluster 0
Womens Smart Casual Shoes	Cluster 0
Mens Goretex	Cluster 0
Mens Formal Shoes	Cluster 0
Mens Casual Shoes	Cluster 0
Mens Boots	Cluster 0
Free Gift	Cluster 0
Acc Handbags and Purses	Cluster 0
Mens Sandals	Cluster 0
Shoe Care	Cluster 1
Womens Formal Sandals	Cluster 2
Womens Casual Shoes	Cluster 3
Womens Casual Boots	Cluster 4
Womens Casual Sandals	Cluster 5
Womens Active Shoes	Cluster 6
Womens Deck Shoes	Cluster 6

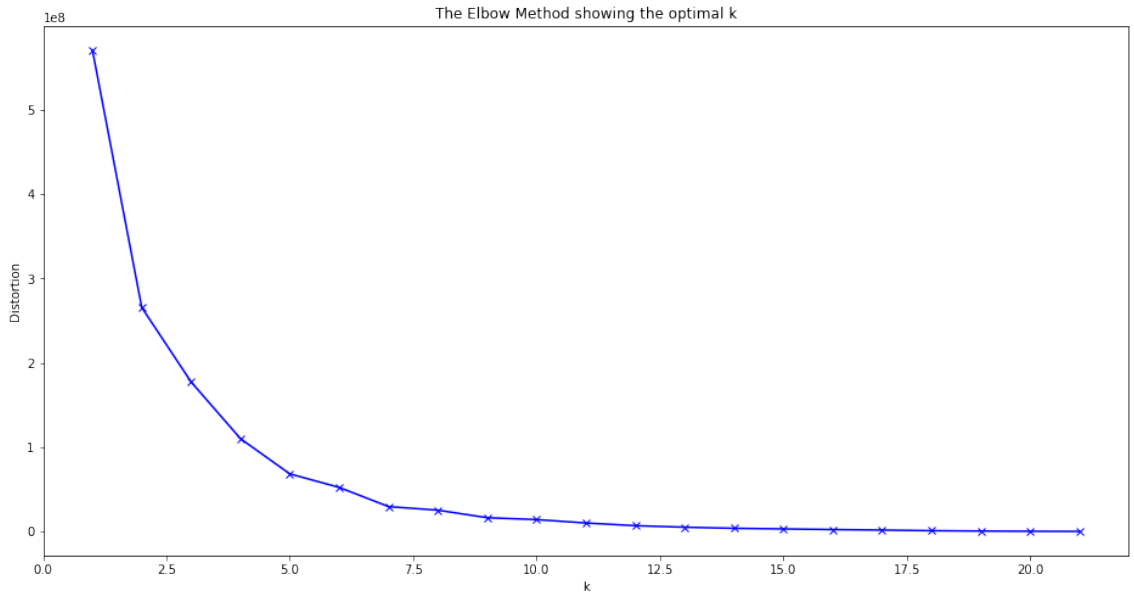

FIGURE A.3: Elbow method plot for Category Clusters

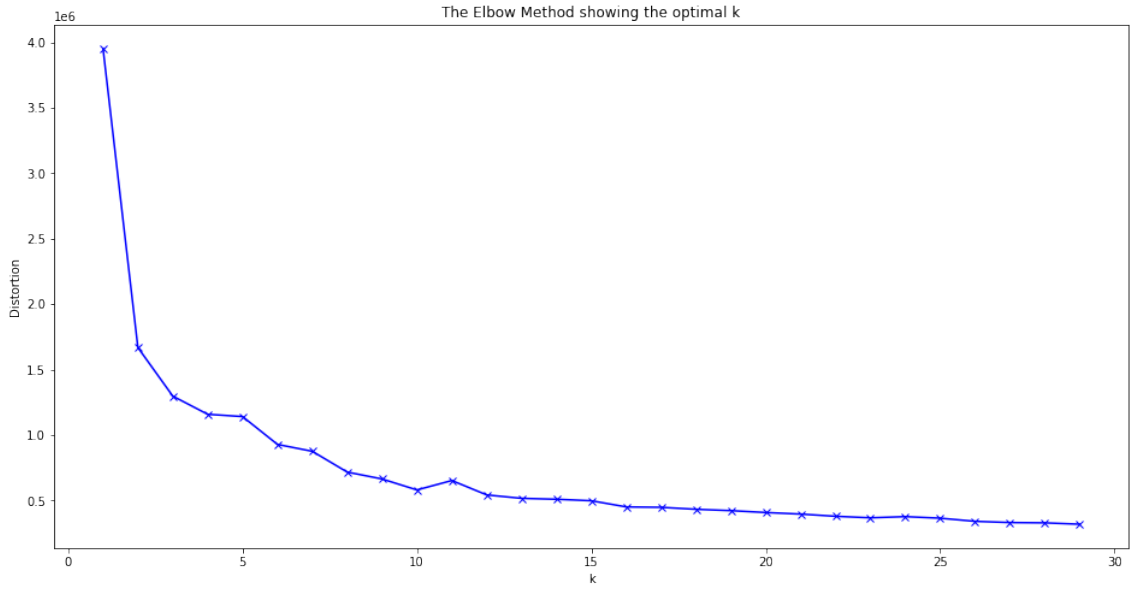

FIGURE A.4: Elbow method plot for Product Clusters

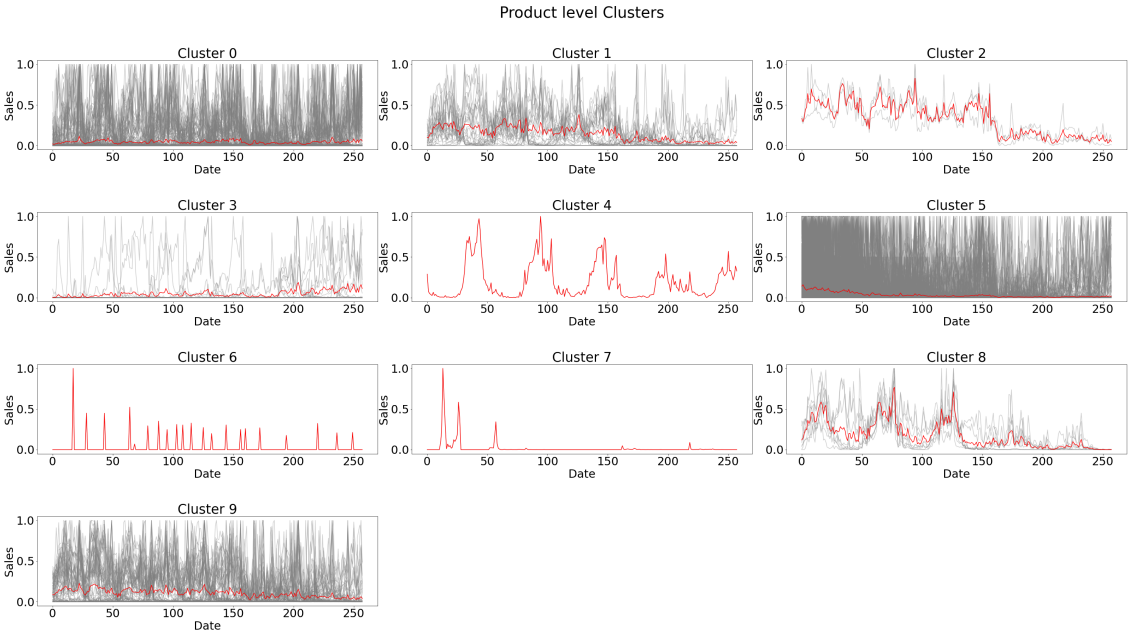

FIGURE A.5: Plots of the different Product Clusters, with the category time series (grey) and the corresponding centroids (red)

Appendix B

Model information

B.1 Custom Model

B.1.1 Basic Model with Residual Data

We have trained the basic model with the observed data and with the sum of trend and residual components. Results can be found in Figure B.1. We can see that the data points we are evaluating are almost the same and that the obtained results are also extremely similar.

B.1.2 FAM results for other products

In Table B.1 we can see the metrics for a subset of products using the FAM for training and for test.

B.2 CatBoost and Linear Regression Model

We have compared three different products from Womens Goretex category. In Figures B.2, B.3 and B.4 we can find the modelling and forecasting of the sales for the products Ridge, Mist and Crest, respectively. For the first product we have data from 2019 to 2021, for the second from 2017 to 2021 and the third one has only been sold during 2021. Metrics can be found on Table B.2.

TABLE B.1: Metrics for FAM for subset of products.

Product	Experiment	RSME	MAE	MAPE
Suede And Nubuck Brush	Training	2810.50	42.38	0.19
	Test	4223.51	54.03	0.25
H Women's Insoles	Training	5034.99	57.61	0.20
	Test	3034.73	44.69	0.15
Renovating Cream	Training	68602.58	205.65	0.20
	Test	57684.17	204.90	0.18
Wax Oil	Training	195.21	10.44	0.40
	Test	790.47	25.15	0.45

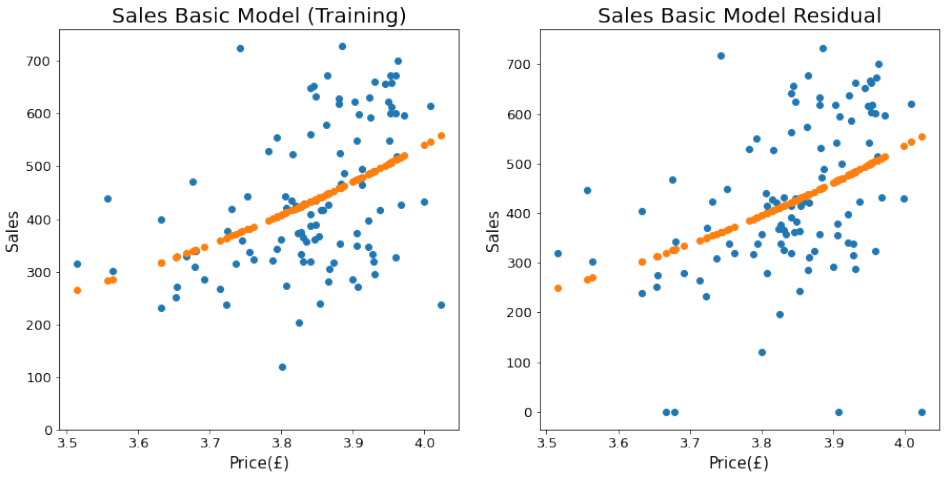

FIGURE B.1: Plots of the Basic model trained by observed data (left) and trained by residual + trend (right)

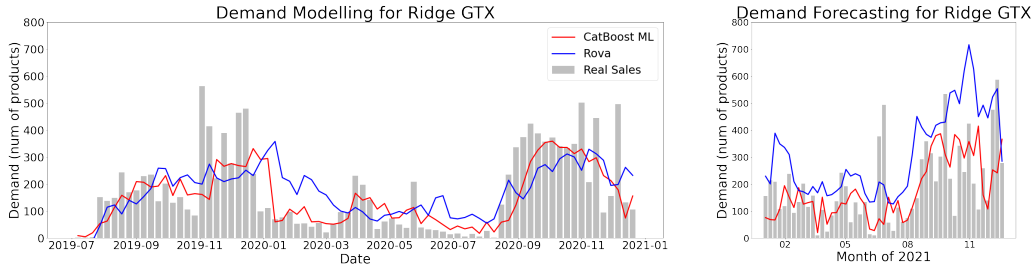

FIGURE B.2: Moddeling and prediction for Ridge product using CatBoost (red) and ROVA (blue).

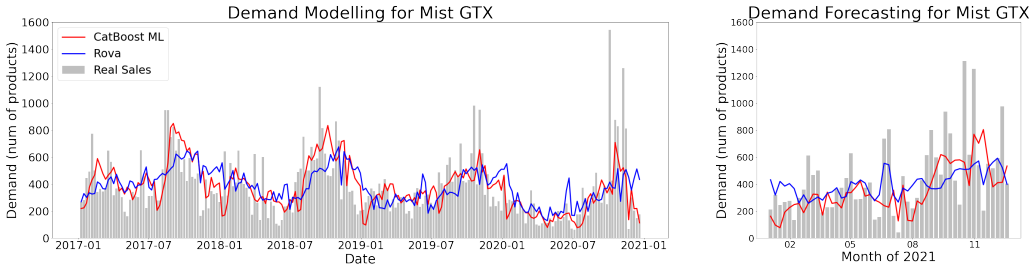

FIGURE B.3: Moddeling and prediction for Mist product using CatBoost (red) and ROVA (blue).

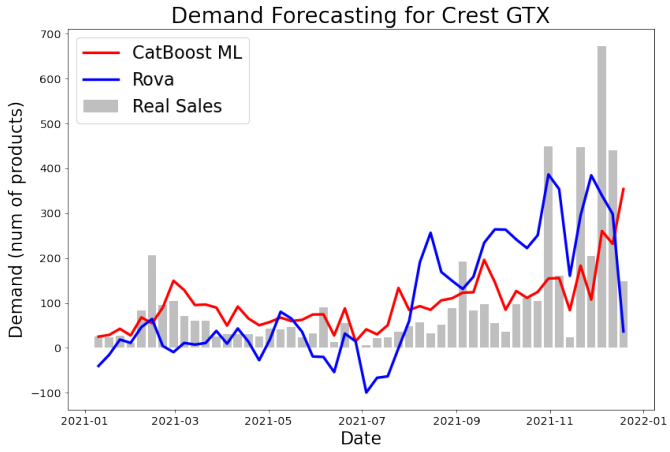

FIGURE B.4: Moddeling and prediction for Crest product using CatBoost (red) and ROVA (blue).

TABLE B.2: Metrics for all the studied models for the product Footwear Cleaner

Product	Experiment	Model	RSME	MAE	MAPE
Ridge	Training	CatBoost	10847.15	70.26	0.93
		ROVA	16451.06	102.37	2.37
	Test	CatBoost	17885.29	90.82	0.63
		ROVA	33807.59	145.81	1.83
Mist	Training	CatBoost	34963.70	132.86	0.40
		ROVA	40969.25	146.52	0.48
	Test	CatBoost	66521.91	197.28	0.50
		ROVA	57518.70	175.57	0.52
Crest	Test	CatBoost	10466.74	62.32	1.18
		ROVA	12728.91	88.61	2.09

TABLE B.3: Top 10 variables interactions

Feature 1	Feature 2	Interaction Strength
prev_sales2	Price	6.609174
Price	ProductCluster	5.943834
prev_sales2	prev_sales4	4.247279
prev_sales4	Price	4.157602
Price	Discount	3.501811
Price	Cost	3.500157
prev_sales2	ProductCluster	3.059588
Week_sin	Price	2.711374
prev_sales2	prev_sales3	2.679977
prev_sales3	Price	2.660856

B.3 CatBoost explainability

In Figure B.5 we can see the feature importance of the variables used in the CatBoost model. In Table B.3 we can see the top ten interactions between variables.

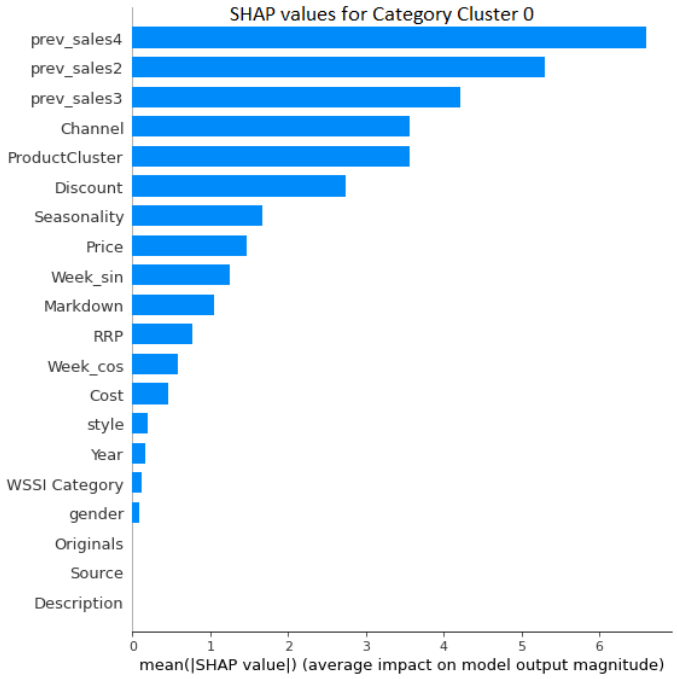

FIGURE B.5: Feature relevance obtained by the TreeShap functionality.

Bibliography

Alon, Ilan, Min Qi, and Robert Sadowski (May 2001). "Forecasting aggregate retail sales: A comparison of artifcial neural networks and traditional methods". In: *Journal of Retailing and Consumer Services* 8, pp. 147–156. DOI: 10.1016/S0969-6989(00)00011-4.

Alston, Julian, James Chalfant, and Nicholas Piggott (June 2002). "Estimating and Testing the Compensated Double-Log Demand Model". In: *Applied Economics* 34, pp. 1177–86. DOI: 10.1080/00036840110086003.

Antipov, Evgeny and Elena Pokryshevskaya (Oct. 2020). "Interpretable machine learning for demand modeling with high-dimensional data using Gradient Boosting Machines and Shapley values". In: *Journal of Revenue and Pricing Management* 19. DOI: 10.1057/s41272-020-00236-4.

Autograd Mechanics (n.d.). https://pytorch.org/docs/stable/notes/autograd.html. Accessed: 2022-06-17.

Bentéjac C., Csörgő A. Martínez-Muñoz G. (2021). "A comparative analysis of gradient boosting algorithms." In: *Artif Intell Rev* 54, 1937–1967. URL: https://doi-org.sire.ub.edu/10.1007/s10462-020-09896-5.

Fildes, Robert, Shaohui Ma, and Stephan Kolassa (2019). "Retail forecasting: Research and practice". In: *International Journal of Forecasting*. ISSN: 0169-2070. DOI: https://doi.org/10.1016/j.ijforecast.2019.06.004. URL: https://www.sciencedirect.com/science/article/pii/S016920701930192X.

Huber, Jakob and Heiner Stuckenschmidt (2020). "Daily retail demand forecasting using machine learning with emphasis on calendric special days". In: *International Journal of Forecasting* 36.4, pp. 1420–1438. ISSN: 0169-2070. DOI: https://doi.org/10.1016/j.ijforecast.2020.02.005. URL: https://www.sciencedirect.com/science/article/pii/S0169207020300224.

Hyndman R.J., Athanasopoulos G. (2018). "Forecasting: Principles and Practice, 2nd edition". In: *OTexts: Melbourne, Australia*. URL: https://otexts.com/fpp2/classical-decomposition.html.

Jain, Anil K. (2010). "Data clustering: 50 years beyond K-means". In: *Pattern Recognition Letters* 31.8. Award winning papers from the 19th International Conference on Pattern Recognition (ICPR), pp. 651–666. ISSN: 0167-8655. DOI: https://doi.org/10.1016/j.patrec.2009.09.011. URL: https://www.sciencedirect.com/science/article/pii/S0167865509002323.

Lingelbach, Katharina et al. (2021). "Demand Forecasting Using Ensemble Learning for Effective Scheduling of Logistic Orders". eng. In: *Advances in Artificial Intelligence, Software and Systems Engineering*. Lecture Notes in Networks and Systems. Cham: Springer International Publishing, pp. 313–321. ISBN: 3030806235.

Lopienski, Kristina (2019). "What is Demand Forecasting? Importance and Benefits of Forecasting Customer Demand". In: *Review of Scientific Instruments*. URL: https://www.shipbob.com/blog/demand-forecasting/.

Lundberg, Scott M. et al. (2020). "From local explanations to global understanding with explainable AI for trees". eng. In: *Nature machine intelligence* 2.1, pp. 56–67. ISSN: 2522-5839.

Ma, Shaohui, Robert Fildes, and Tao Huang (2016). "Demand forecasting with high dimensional data: The case of SKU retail sales forecasting with intra- and inter-category promotional information". In: *European Journal of Operational Research* 249.1, pp. 245–257. ISSN: 0377-2217. DOI: https://doi.org/10.1016/j.ejor.2015.08.029. URL: https://www.sciencedirect.com/science/article/pii/S0377221715007845.

Madhulatha, T. Soni. (2012). "An overview on clustering methods". In: *IOSR Journal of Engineering7*, pp. 719–725. URL: https://arxiv.org/abs/1205.1117.

Marshall, Alfred (1949). "Principles of Economics:An Introductory Volume". In.

Molnar, Christoph, Giuseppe Casalicchio, and Bernd Bischl (2021). "Interpretable Machine Learning – A Brief History, State-of-the-Art and Challenges". eng. In: *ECML PKDD 2020 Workshops*. Communications in Computer and Information Science. Cham: Springer International Publishing, pp. 417–431. ISBN: 9783030659646.

Nielsen, Frank (Feb. 2016). "Hierarchical Clustering". In: pp. 195–211. ISBN: 978-3-319-21902-8. DOI: 10.1007/978-3-319-21903-5_8.

Pearl, Judea, Madelyn Glymour, and Nicholas P. Jewell (2016). "Causal Inference in Statistics: A Primer". eng. In: 88.

Prokhorenkova L Gusev G, Vorobev A-Dorogush AV Gulin A (2018). "Catboost: unbiased boosting with categorical features." In: *arXiv:1706.09516*. URL: https://arxiv.org/abs/1706.09516.

Senin, Pavel (Jan. 2009). "Dynamic Time Warping Algorithm Review". In.

Smirnov, P S and V A Sudakov (2021). "Forecasting new product demand using machine learning". eng. In: *Journal of physics. Conference series* 1925.1, pp. 12033–. ISSN: 1742-6588.

Syakur, M A et al. (2018). "Integration K-Means Clustering Method and Elbow Method For Identification of The Best Customer Profile Cluster". In: *IOP Conference Series: Materials Science and Engineering* 336, p. 012017. DOI: 10.1088/1757-899x/336/1/012017. URL: https://doi.org/10.1088/1757-899x/336/1/012017.

Thomassey, Sébastien and Antonio Fiordaliso (2006). "A hybrid sales forecasting system based on clustering and decision trees". In: *Decision Support Systems* 42.1, pp. 408–421. ISSN: 0167-9236. DOI: https://doi.org/10.1016/j.dss.2005.01.008. URL: https://www.sciencedirect.com/science/article/pii/S0167923605000114.

Tian, Xin, Haoqing Wang, and Erjiang E (2021). "Forecasting intermittent demand for inventory management by retailers: A new approach". eng. In: *Journal of retailing and consumer services* 62, pp. 102662–. ISSN: 0969-6989.

Ulrich, Matthias et al. (2022). "Classification-based model selection in retail demand forecasting". In: *International Journal of Forecasting* 38.1, pp. 209–223. ISSN: 0169-2070. DOI: https://doi.org/10.1016/j.ijforecast.2021.05.010. URL: https://www.sciencedirect.com/science/article/pii/S0169207021000935.