# A machine learning approach to promotional sales forecasting

*Pedro Schuller de Almeida e Graça Barbosa*

**Master's Dissertation**

Supervisor: Prof. José Luís Moura Borges

U.PORTO

FEUP FACULDADE DE ENGENHARIA
UNIVERSIDADE DO PORTO

**Mestrado Integrado em Engenharia e Gestão Industrial**

2018-02-01

# Abstract

Grocery retail is one of the most competitive industries in Portugal and in the World. In an economy of ever growing complexity, fiercer competition, and higher consumer standards, enterprises must strive to improve their value proposition and the quality of their processes to stay ahead. When retailers are faced with the challenge of providing customers with quality products and high service levels, while managing at the same time to maintain low resource usage, sales forecasting is endowed with particular relevance. Demand is more unpredictable than ever, and while overestimating it will force selling products at a markdown price and increase handling and shrinkage costs, under-stocking will damage both a company's reputation and sales.

This thesis arises in the context of an ongoing project with a national leader in food retail, with the objective of improving the replenishment capabilities for promotional campaigns at the retailer's Fresh Food Division. The scope of this project adds to the arduousness of the task, both because promotional sales are harder to predict, and because the costs associated with forecasting error aggravate in the presence of products with a reduced shelf-life.

More specifically, this work addresses the forecasting methodology currently in place. The implementation of a previously developed algorithm revealed inherent limitations about its construction. Such algorithm was optimized for a specific product line, and its predictive abilities have not endured when applied to different categories.

The work related to this thesis began by mapping the current methodology in place at the retailer, as well as with an extensive review on modern retail sales forecasting models and machine learning algorithms and techniques. The respective findings were put into practice through the formulation of four algorithms for promotional sales forecasting. To have a reliable estimate of each algorithm's potential, roughly a year and a half of sales data regarding three product lines with distinctive characteristics were selected and assumed to represent the whole Fresh Division. A framework was developed in order to optimize and select the best model for each of the three categories. Lastly, this thesis describes the development and implementation of a methodology to include product interaction, i.e., sales cannibalization and complementarity, in the sales forecasting models.

The algorithms developed show a significant improvement in forecasting accuracy. The implementation of these new methodologies is expected to reduce inventory levels and improve in-stock rates at the retailer's stores. Moreover, it was shown that the ideal algorithm is highly dependent on the sales patterns. Additionally, the results obtained do not strongly support the thesis whether the sales of a given product can be partly explained by promotional activity in other products from the same category.

# Resumo

A indústria do retalho alimentar é uma das mais competitivas a nível nacional e mundial. A crescente complexidade da economia, acompanhada por um aumento de competição e dos critérios dos consumidores, obriga as empresas a refinar a sua proposta de valor e a optimizar os seus processos para se manterem no mercado. Quando grandes retalhistas se deparam com o desafio de garantir altos níveis de serviço ao cliente, ao mesmo tempo mantendo os custos a um nível aceitável, a previsão de vendas ganha um papel de destaque. A procura apresenta-se mais incerta do que nunca, e enquanto a sua sobrestimação resultará em níveis de inventário excessivos e reduções de preço forçadas, a insuficiência de stock em loja tem como consequência vendas perdidas e perda de reputação junto do cliente.

Esta dissertação surge no contexto de um projecto em conjunto com um retalhista líder a nível nacional, que tem como propósito a melhoria do reaprovisionamento promocional na Direcção Comercial de Perecíveis. Este foco representa uma dificuldade e relevância acrescida já que, para além do facto de vendas promocionais serem mais difíceis de prever, os custos logísticos associados ao erro de previsão são mais elevados, dada a perecibilidade dos produtos em causa.

Este trabalho aborda, especificamente, a metodologia de previsão actualmente em funcionamento. A implementação de uma metodologia previamente desenvolvida revelou algumas das suas limitações inerentes. Por exemplo, o facto de o anterior algoritmo ter sido desenvolvido e optimizado para uma linha de produto em específico poderá explicar o seu inferior sucesso quando transferido para outras categorias.

As actividades desenvolvidads no decurso desta tese começaram por um levantamento dos processos actualmente em vigor no retalhista, por uma extensa revisão teórica sobre modelos de previsão modernos assim como algoritmos e técnicas de aprendizagem automática. As respectivas descobertas foram postas em prática através da formulação de quatro algoritmos de previsão de vendas promocional. Por forma a obter uma estimativa robusta do potencial de cada algoritmo, cerca de um ano e meio de registos de vendas promocionais que dizem respeito a três categorias com características distintas entre si foram seleccionados e assumidos com representativos da Direcção Comercial de Perecíveis como um todo. Adicionalmente, foi desenvolvido um procedimento para optimizar e seleccionar o melhor método para cada categoria. Por fim, esta tese descreve o desenvolvimento e implementação de uma metodologia que visa incluir a interacção entre produtos, isto é, a existência de fenómenos de substituição e complementaridade, na previsão de vendas.

Os algoritmos desenvolvidos apresentam melhorias significativas no que toca à precisão da previsão de vendas promocionais. Espera-se que a implementação das metodologias propostas reduza os níveis de inventário do retalhista e melhore os níveis de serviço ao cliente. Os resultados obtidos sugerem que o algoritmo ideal depende em grande medida dos dados de entrada. Sugerem ainda que as condições de venda promocionais de outros artigos não contêm informação que possa explicar significativamente variações nas vendas de um determinado artigo da mesma categoria.

# Acknowledgements

This thesis concludes my Masters in Industrial Engineering and Management at FEUP. It closes a five-and-a-half year chapter of my life, for which I am most grateful.

I'd like to begin by thanking LTPlabs for welcoming me and providing me with this exciting challenge. All my expectations have been surpassed. LTPlabs has proven to be, apart from an outstanding enterprise, an excellent school as well. I had the pleasure of working closely with Eng. Daniel Pereira, my technical coordinator, and would like to thank him for his patience in teaching me the intricate elements of this project and his availability to revise my work. I was also very fortunate to be able to rely on Eng. Bruno Batista, and I thank him for having readily clarified my doubts and discussed ideas with me. Lastly, I'd like to praise Dr. Teresa Bianchi de Aguiar, my manager and supervisor at LTPlabs, for her tireless spirit in aiding me throughout the project.

It will not be easy to say goodbye to the place that was my second home for more than five years. My passage through FEUP has given me the best years so far, because I got to make good friends, experience the student life and grow as a person. My love for engineering grew under the guidance of the notable professors of this institution. For this, I'd like to particularly thank Professors Armando Leitão, Paulo Tavares de Castro, Álvaro Rodrigues, Bernardo Almada-Lobo, José Faria, Pedro Amorim and Ana Camanho. Special thanks go to my supervisor, Professor José Luís Borges, for his guidance and help, without which my thesis would have been a lot harder to understand.

My student path was deeply marked by my undertakings in juvenile associativism. My mandates at the head of AGE-i-FEUP and ESTIEM have been by far the most enriching experiences of my student life. I have seen the potential of the student's community both here in Porto and in Europe, and it assured me of the bright future ahead. I had the pleasure of sharing these adventures with my dearest 27$^{th}$ Board of ESTIEM, Tiago, Taavi, Rebekka and Aytaç, and here at home with my comrades Miguel Faria and Carlos Guilherme.

I'd like to send a strong hug to my lifelong and well-spoken friend Diogo Cardoso for his willingness to revise my work.

Of course, none of this would have been possible without an incomparable support back at home. From the bottom of my heart, I dedicate this work to my mother Maria João, my father Luís, my brother Tomás, my grandfather Armando and grandmother Christa.

During the course of this thesis, I've discovered that people are much like machine learning algorithms. Our brains work like a pile of linear algebra - we're given inputs like what we see and touch, our neurons fire, and an output is given, like a sentence or a smile. We learn from experience, and try to maximize an objective function, like happiness (if only it was that easy!). A technique often used to improve performance is that of the ensemble, where the predictions of several weak learners are combined to create a strong one. Not being much of a strong learner myself, I've decided to form an ensemble with a special someone. I love you, Ana.

*"In God we trust, all others bring data."*

W. Edwards Deming

# Contents


**1 Introduction** **1**
- 1.1 Company description . . . 2
- 1.2 Motivation . . . 2
- 1.3 Shortcomings of the current methodologies . . . 3
- 1.4 Approach . . . 4
- 1.5 Scope . . . 4
- 1.6 Goals . . . 6
- 1.7 Thesis Outline . . . 6

**2 Theoretical Background** **7**
- 2.1 Challenges in forecasting in grocery retail . . . 7
- 2.2 Forecasting models . . . 8
  - 2.2.1 An overview on forecasting models . . . 8
  - 2.2.2 Measures of accuracy . . . 9
  - 2.2.3 Linear Models and the SCAN*PRO . . . 10
  - 2.2.4 Base-times-Lift Approaches . . . 11
  - 2.2.5 Random Forests . . . 11
  - 2.2.6 Gradient Boosting Machine . . . 12
  - 2.2.7 Other machine learning concepts . . . 13
- 2.3 Factors that influence demand . . . 16
  - 2.3.1 Time series and events . . . 16
  - 2.3.2 Store . . . 17
  - 2.3.3 Product . . . 17
  - 2.3.4 Promotions . . . 18
- 2.4 Product interaction . . . 19

**3 Problem description** **21**
- 3.1 Current methodology . . . 21
  - 3.1.1 Available Data . . . 21
  - 3.1.2 Solution . . . 23
  - 3.1.3 Variables used . . . 24
  - 3.1.4 Model formulation . . . 25
  - 3.1.5 Forecasting and replenishment procedure . . . 27
- 3.2 Limitations . . . 28

**4 Methodology** **31**
- 4.1 Approach . . . 31
- 4.2 Inputs . . . 32
  - 4.2.1 Sales Baseline . . . 32

4.2.2 Variables . . . 33
4.2.3 Training and test datasets . . . 33
4.3 Formulations . . . 34
4.4 Evaluation and Tuning . . . 36
4.5 Product Interaction . . . 37

**5 Results** **39**
5.1 Forecasting results . . . 39
5.2 Properties of the models obtained . . . 43
5.3 Visualization . . . 47

**6 Conclusion** **49**
6.1 Implications for practice . . . 49
6.2 A reflection on forecasting culture . . . 50
6.3 Future research . . . 51

**A The constant replenishment model** **57**

**B Model tuning and selection procedure** **59**

**C Algorithm graphical outputs** **61**

**D Dashboard** **65**

# Acronyms and Symbols

| | |
|---|---|
| AIC | Akaike Information Criterion |
| CART | Classification and Regression Tree |
| CV | Coefficient of Variation |
| EBITDA | Earnings Before Interest, Taxes, Depreciation and Amortization |
| ERP | Enterprise Resource Planning |
| ETL | Extract, Transform and Load |
| GBM | Gradient Boosting Machine |
| IS | Information System |
| KPI | Key Performance Indicator |
| LASSO | Least Absolute Shrinkage and Selection Operator |
| LM | Linear Model |
| MAPE | Mean Absolute Percentage Error |
| MPE | Mean Percentage Error |
| RDF | Retail Demand Forecasting |
| RMS | Retail Management System |
| SKU | Stock Keeping Unit |

# List of Figures


1.1 The retailer's structure, Fresh Division in detail . . . 2
1.2 Breakdown of the cost of forecasting error according to Kahn (2003) . . . 3
1.3 The retailer's Product Structure . . . 5
1.4 The retailer's Store Structure . . . 5

2.1 Classification of Forecasting Models (in: Gonçalves (2000)) . . . 9
2.2 Plots of 4 models obtained via fitting polynomials of different orders M . . . 14
2.3 A visual interpretation of regularization methods . . . 15

3.1 Price cut and Communication Means relative importance . . . 22
3.2 Forecasting and Replenishment solution schematics . . . 23
3.3 An example of sales correction given a product shortage . . . 24
3.4 Forecasting and Replenishment procedure for a specific promotional week . . . 27
3.5 Weekly Sales by Discount Type . . . 29

4.1 Illustrative timeline of the project . . . 31

5.1 GBM initial results with respect to its hyperparameters . . . 40
5.2 Sales and Forecast comparison for a given product . . . 43
5.3 Sales and Forecast scatter plot . . . 43
5.4 Sales response of a given product to its continuous input variables . . . 44
5.5 Sales response of a given product to some of its categorical input variables . . . 44
5.6 Evolution of MAPE for the Fruits category . . . 45
5.7 Relative variable importance in the Fruits Category . . . 45
5.8 Example of a product interaction plot . . . 46
5.9 Forecasting indicator panel of the dashboard . . . 47

A.1 The constant replenishment model . . . 57

C.1 Evolution of MAPE for the Chicken category . . . 61
C.2 Coefficients of the continuous variables in the Chicken category . . . 62
C.3 Variable importance of the categorical variables in the Chicken category . . . 62
C.4 Evolution of MAPE for the Frozen Fish category . . . 63
C.5 Coefficients of the continuous variables in the Frozen Fish category . . . 63
C.6 Variable importance of the categorical variables in the Frozen Fish category . . . 64

D.1 Overall dashboard structure . . . 65
D.2 Forecasting indicator pane . . . 66
D.3 Replenishment indicator pane . . . 67
D.4 Weekly indicator profile . . . 68

D.5 Overforecasting and stock control . . . . . 69
D.6 Underforecasting and stock-out control . . . . . 70
D.7 Detailed week view . . . . . 71
D.8 The service level - stock level trade-off . . . . . 72

# List of Tables


3.1 Results of the current model on the selected test period . . . . . 29

4.1 MAPE and Bias of the baseline forecast in the test period . . . . . 32
4.2 Number of observations per category . . . . . 33
4.3 Hyperparameters for each model . . . . . 37

5.1 Results . . . . . 41
5.2 Results after including product interaction . . . . . 42
5.3 Comparison of results and estimate of business impact . . . . . 42

# Chapter 1

# Introduction

Grocery retail, as any industry that provides goods essential for living, is highly competitive. Companies use increasingly complex and aggressive methods to compete for a share of the population's income, at thin profit margins. A market researcher (BDO (2016)) claims in its report that, in Portugal, grocery retailers operate, on average, at 3% EBITDA margins and 17% gross margins. Moreover, it is a highly concentrated industry, a fact that can be explained by large economies of scale. INE (2017) reports that in a universe of €12.1 billion in sales of goods, 78.6% of all sales volume circulates through the top percentile of retail enterprises, in terms of size. In 2016, there were 133,000 enterprises registered under "retail", 1.5% less than 2015, which indicates that concentration is likely to increase. As a consequence, retailers tend to have an increasing bargaining power with local and national producers. In 2016, sales of own brand products already represented 34.4% of the total sales of food.

The increasing complexity of operations, fiercer competition, and higher consumer standards pose new challenges for retailers to address. In such an environment, efficient operations and parsimonious resource usage, on the one hand, and the ability to deliver value to the end customer, on the other, become all the more relevant. When it comes to delivering the right products to the right place, in ideal conditions and at the lowest cost possible, accurate demand forecasting and agile yet efficient stock replenishment processes are preponderant.

This thesis arises in the context of an ongoing project with a major Portuguese food retailer. The project consists in the development of improved forecasting and replenishment methodologies for the retailer's supply chain. Its scope is promotional sales in the Fresh Food Division. Promotional sales pose an aggravated challenge for forecasting and replenishment, due to their increased variability and sales volume, at lower unit profit margins. Moreover, the reduced shelf-life of fresh products magnifies the costs associated with forecasting errors and replenishment inefficiencies.

The project has, previously to this thesis, produced a new forecasting algorithm and replenishment procedures, which are currently used by stock managers of the Fresh Division. The underlying methodologies can be found in Batista (2016), and will, alongside the developments and findings made so far, be described in greater detail in Chapter 3.

## 1.1 Company description

The company where this project is being undertaken is a national leader in food retail. The sales volume attributable to its own brands is of around 30%. The company is organised in 3 Divisions, each containing several Business Units (e.g. Fruits and Vegetables within the Fresh Division) as seen in Figure 1.1. The retailer has stated as one of its main objectives to be recognised as a specialist in perishable items, and has made efforts to improve and better communicate its value proposition, as well as improve the quality of related processes, to which this thesis aims to contribute.

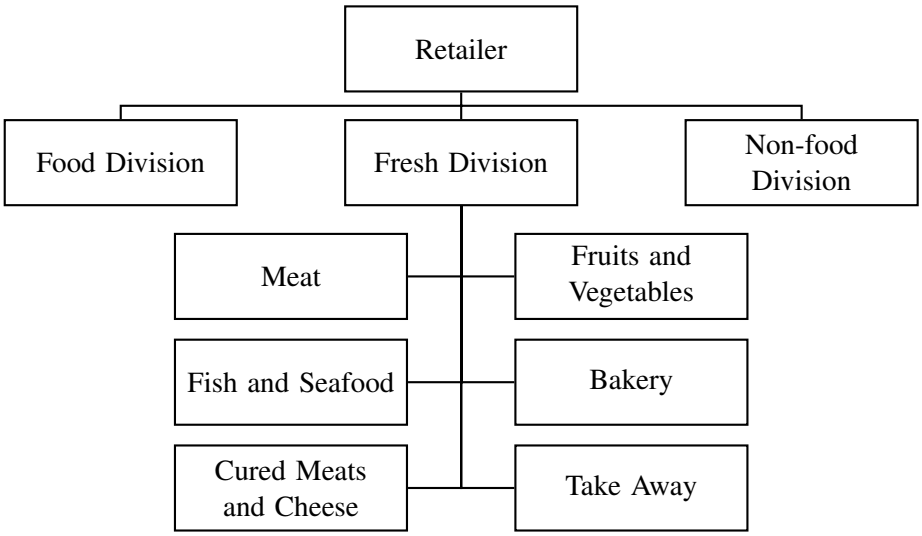



Figure 1.1: The retailer's structure, Fresh Division in detail

## 1.2 Motivation

According to Arunraj and Ahrens (2015), when forecasting sales with respect to availability, the trade-off is between over-stocking and under-stocking. Kahn (2003) systematized the impacts of forecasting error and provided a framework for measuring them. Figure 1.2 further details the types of costs associated with forecasting error.

Furthermore, Kahn (2003) proposes a rule of thumb for estimating the cost of forecasting error: The cost of additional 1% of over-forecasting error of an SKU equals $1\% * SKU\ Volume * (unit\ cost + holding\ cost)$ and that of under forecasting equals $1\% * SKU\ Volume * unit\ profit\ margin$. Of course, the global cost of forecasting error would likely be less than the sum of these two costs, due to the combination of under- and over-forecasting situations.

We can conclude that small gains in forecasting accuracy in any product category have significant impacts in a retailer's net profits. Therefore, minimizing the inevitable forecasting error mitigates the negative consequences of both under- and over-forecasting.

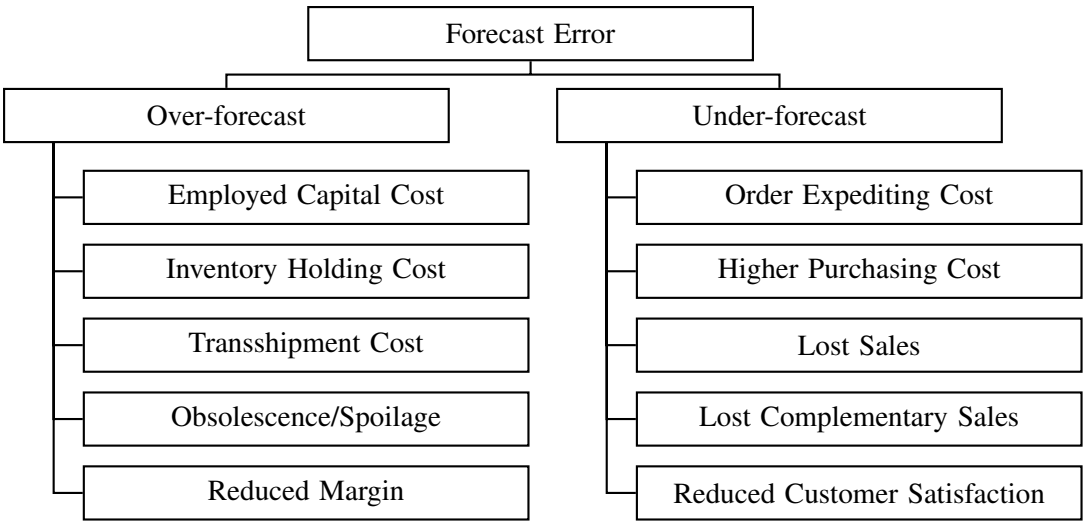



Figure 1.2: Breakdown of the cost of forecasting error according to Kahn (2003)

Apart from the business upside of improving sales forecasts, one must not forget the "moral cost", so to speak, of allowing food to expire and be thrown away. Within the next decade the European Parliament wants to reduce food waste by 50% and improve access to food for needy citizens (see Donselaar et al. (2016)).

## 1.3 Shortcomings of the current methodologies

The project's unfolding has, in the eyes of the retailer company, added value to its forecasting and replenishment processes. For instance, a more conservative replenishment has reduced excess stock at the end of promotional weeks, even when sales on weekends have increased. From the process point of view, predictions are now made at a higher level of detail, and follow the same procedures across all Business Units.

However, the promising results obtained by Batista (2016) in terms of forecasting accuracy in the Fruits category have not been transposed to other product lines with the same success, nor have persisted with time. This lead to the suspicion that the algorithm developed was able to produce exceptional results for that particular time period and product line only. Batista (2016) also developed a product cannibalization methodology which is, however, not currently in use, since it showed no ability in practice to add value to the forecasts.

## 1.4 Approach

To tackle the identified shortcomings, an extensive review was made on machine learning algorithms, best practices in machine learning, and product cannibalization. Afterwards, according to the findings of the aforementioned literature review, several algorithms will be developed iteratively and evaluated in a way that allows the project team to be confident that the obtained results will have a more permanent nature.

## 1.5 Scope

The selection of stores and product lines to be considered was made according to the retailer's two hierarchical structures - a product structure and a store structure, which are detailed in Figures 1.3 and 1.4, respectively. In an attempt to have a selection of products that would be representative of the whole Division, but that would not be overwhelming in terms of dataset size, the models developed will be validated with sales data of three distinct categories:

1. The Fruits category contains $\sim$500 SKUs and is the most perishable category of all, with shelf-lives as low as 3 days, highly aggravating the cost of over-forecasting. Successfully capturing seasonality is essential, as many of its products have extremely seasonal sales patterns.

2. The Chicken category has $\sim$200 SKUs and is a category that has generally good replenishment indicators (both high service levels and low stock coverage).

3. The Frozen Fish category contains $\sim$250 SKUs and is characterized by very high stock levels, alongside a very high shelf life (> 1 year, in some cases).

These categories contain about 9% of all SKUs of the Fresh Division. The stores that will integrate the analysis are of the following formats:

**Format A** Large stores with a wide range of products, located in areas of high population density and customer traffic.

**Format B** Medium sized stores, usually outside major cities.

**Format C** Smaller stores located to serve specific population centres.

Only the stores of mainland Portugal will be accounted for, as well as only these three main store formats, which totals 245 establishments. Promotional sales forecasts are made at the SKU/-Store Format/Week level, and then distributed through the week via weekday indexes. As a consequence, the forecasting accuracy metric is done at the same weekly level.

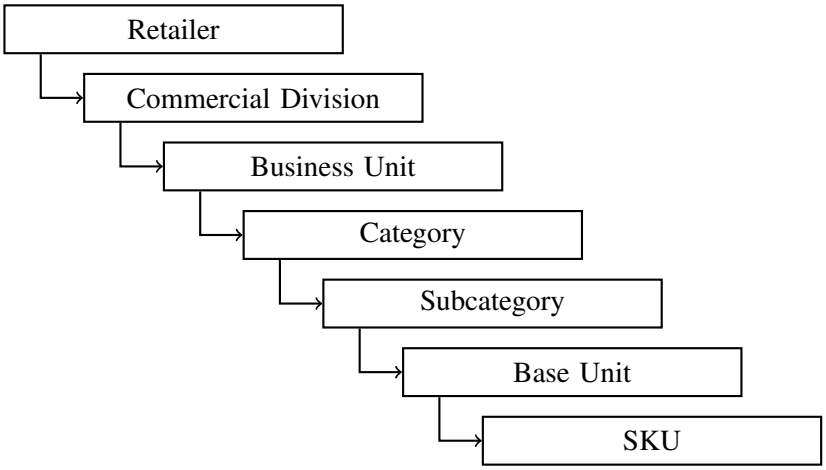



Figure 1.3: The retailer's Product Structure

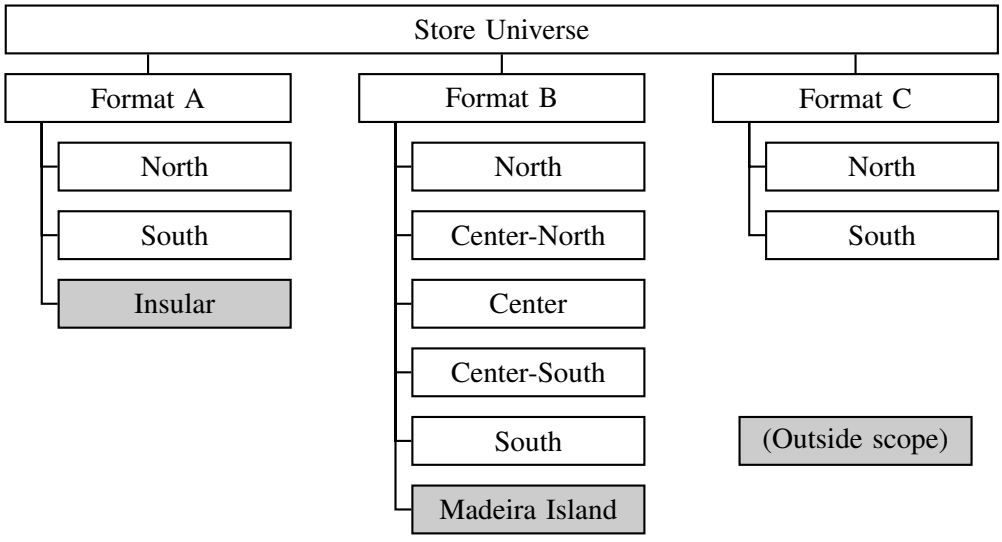



Figure 1.4: The retailer's Store Structure

## 1.6 Goals

This thesis is expected to produce one or several models that are significantly better performing[1] than the one currently in use for forecasting promotional sales in the retailer's Fresh Division. As secondary objectives, this work also aims to evaluate the applicability and performance of machine learning algorithms and techniques when applied to real world scenarios, to identify interactions between products (cannibalization) within the same category, and to improve the visibility of business indicators among the forecasting teams.

## 1.7 Thesis Outline

This thesis is organised in six chapters. Its outline is as follows:

Chapter 1 introduces the thesis, where the problem is contextualized and defined and the main objectives are stated.

Chapter 2 aims to provide theoretical background on the relevant subjects for this work. Firstly, the characteristics of the grocery retail industry are described. Secondly, a literature review on forecasting models and machine learning algorithms and concepts is conducted. Lastly, the typical factors considered as relevant for sales forecasting in retail are enumerated and explained.

Chapter 3 describes in detail the current data, processes and methodologies in use at the Fresh Division for forecasting sales, as well as their performance and limitations.

Chapter 4 briefly describes the timeline of the work related with this thesis. It identifies the new variables used as inputs by the models. Afterwards, the process of construction of the several models used is described, as well as the techniques used to tune these models to optimize performance. Lastly, a methodology for integrating product interaction is characterized.

Chapter 5 begins to show the performances of the models obtained. The properties of these models are described in the second part of this chapter. Lastly, this chapter describes a dashboard developed and currently in use for the forecasting teams to evaluate their results.

Chapter 6 is a summary and a reflection on the findings of this thesis.

[1] According to the metrics defined in Chapter 2.

# Chapter 2

# Theoretical Background

## 2.1 Challenges in forecasting in grocery retail

As mentioned in the introductory chapter of this thesis, the grocery retail industry is highly competitive, works on low margins, and deals with the very complex task of balancing under- with over-stocking. Academia has further arguments to why demand forecasting in the retail context is not a trivial ordeal, and some of them will be presented below.

**Increase in promotional activity** In an interview to a Portuguese grocery retail industry executive, Miranda (2017) reports that promotions represent an increasing portion of sales, a figure estimated to be around 45% in Portugal. New, less experienced players are now entering the market, experimenting new tactics and methodologies, and allegedly introducing unhealthy tendencies in the industry. Outside Portugal, there has also been an increase of promotional intensity - over the last 4 years, between 10 to 40%, depending on the product category.

**Increase in assortment size** Dekker et al. (2004) have shown that the average assortment in supermarkets has grown from around 6000 in the 1990's to an astounding 30,000 in 2004. It is presumed that this figure has continued to increase until this moment. Demand for a set of characteristics is now distributed over a larger assortment of products, creating a sort of portfolio effect that makes predicting quantities of specific products very difficult.

**Decrease in product life cycle** Dekker et al. (2004) further claim that not only did the assortment size increase, there has also been a decrease in the product life cycle time. Manufacturers innovate and adapt to new market trends, such as healthy products, economic sizes, rebranding, repackaging, and special/limited editions. There is, therefore, an increased scarcity of sales data for individual products which, combined with an increase in the number of products, further increases the complexity of constructing reliable forecasts. Huang et al. (2014) have shown that customers are more likely to switch stores and never come back than observed before. They are less willing than ever to either purchase substitutes or delay purchases.

**Perishability** INE 2017 report that, in Portugal, perishable products account for 50% of retailers' total sales volume. Apart from short shelf-lives, these products often also have more demanding storage conditions than their less perishable counterparts. The added undesirability of stocking perishables calls for more frequent replenishments and shorter-term forecasts, the latter having become more difficult in the recent decades, according to Dekker et al. (2004). This need also adds stress to the supply-chain, which may cause further product unavailability at the desired time.

**Human element** No matter how powerful a forecasting algorithm is, its outputs will always need validation to consider the effect of variables the model has not accounted for (such as competition, for example), to predict the effect of new conditions the model has not been trained on, to identify clearly misaligned forecasts, or to replace the model for products/stores with insufficient data. However, validation is resource intensive, which led Kourentzes and Petropoulos (2016) to believe that experts are not able to account for and collect all relevant data to optimally validate their forecasts.

## 2.2 Forecasting models

Forecasting is the art and science of predicting future events. It involves analysing historical data and projecting a future situation. Heizer and Render (2006) state that most approaches use a mathematical model for making base predictions, combined with the good judgement of a manager. Moreover, Heizer and Render (2006) claim that even though our forecasting ability has improved, it has been outpaced by the increasing complexity of the world economy. This section comprises of an overview of forecasting models and measures of forecast accuracy being used in retail sales will be made, as well as detailed descriptions of selected models and developments in machine learning that have contributed to improve those and other models.

### 2.2.1 An overview on forecasting models

Gonçalves (2000) breaks down forecasting methods in subjective or qualitative methods and quantitative methods, as depicted in Figure 2.1. Qualitative methods are often used for long-term forecasts and strategic decisions, since many of the assumptions made in mathematical models break down in the long-term. On the other hand, quantitative models are often superior at short-term sales forecasting.

Traditional sales forecasting methods include the naïve approach, where the forecast equals the last sales data point, moving average, and exponential smoothing. These belong to the time-series model group and are usually used to predict baseline sales. More advanced methods within the time-series are the ARIMA models produced by Box et al. (2015). These models have suffered several extensions to incorporate, for example, seasonality (SARIMA) and external variables (SARIMAX).

Sales of an item under a promotion greatly overwhelm the volume of its regular sales. When it comes to forecasting promotional sales, it is believed that the properties of the promotion hold more information than the moment in time the sale is made on. Therefore, authors of almost all recent literature choose to use causal models to forecast promotional sales, and such are the models to be studied and used over the course of this thesis.

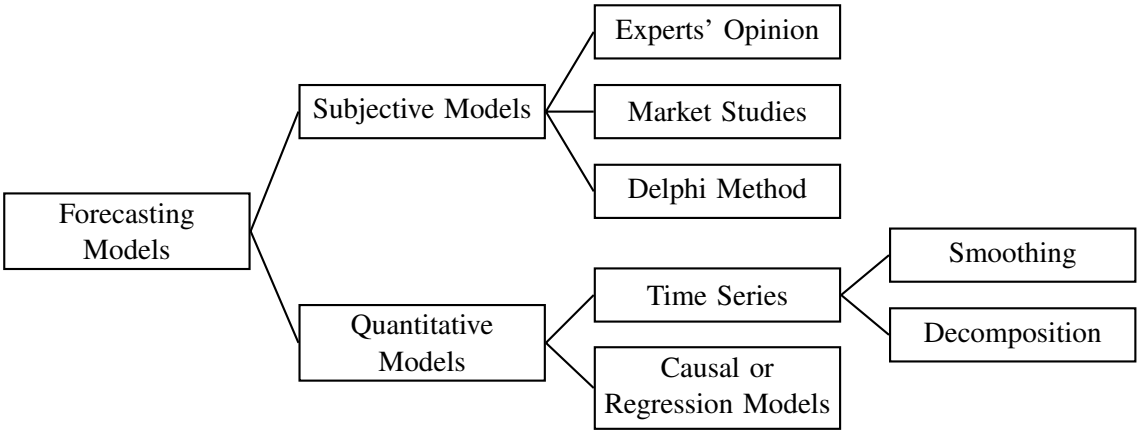



Figure 2.1: Classification of Forecasting Models (in: Gonçalves (2000))

Examples of causal models include: the SCAN*PRO model (that will be described in the next section); a baseline-times-lift approach of Derks (2015), where a multiplying factor for estimated regular sales is estimated out of promotional variables; Ali et al. (2009) use a Support Vector Machine to perform regression; Kuo (2001) uses a more complex machine learning algorithm, called fuzzy neural network, to build rules for promotions. Some authors choose to take the natural log of unit sales, as is the case of Donselaar et al. (2016).

### 2.2.2 Measures of accuracy

The accuracy of any model can be obtained by comparing the forecast values with the actual sales. The most common used metrics are:

**Mean Absolute Deviation or MAD** is the average forecast error of the model and computed by taking the absolute sum of individual forecast errors and dividing by the number of observations: $MAD = \sum_{i=1}^{n} |Forecast - Sales|/n$

**Mean Squared Error or MSE** is the average of the squared differences between forecast and sales: $MSE = \sum_{i=1}^{n} (Forecast\ error^2)/n$ and is used when frequent but small errors are preferred to large infrequent ones.

**Mean Average Percentage Error or MAPE** - a problem with MAD and MSE is that their values depend on the magnitude of the figure being forecast. MAPE takes the average forecast error expressed as a percentage of the actual sales: $MAPE = 100\% \cdot (\sum_{i=1}^{n} |Forecast - Sales|/Sales)/n$

**Mean Percentage Error or MPE** is also called Bias and measures the tendency to either under- or over-forecast demand and equals: $MPE = 100\% \cdot (\sum_{i=1}^{n} (Forecast - Sales)/Sales)/n$

Given their superior interpretability, MAPE and Bias will be used over the course of this thesis as a measure for model quality.

### 2.2.3 Linear Models and the SCAN*PRO

In the case of linear models, a vector of observations $y$ having $n$ components is assumed to be the product of a random variable $Y$, whose components are independently distributed with mean $\mu$. Assuming there is a set of independent variables $X$ that cause $Y$, a linear model may be written:

$$E(Y_i) = \mu_i = \sum_{1}^{p} x_{ij}\beta_j; \qquad i = 1, \ldots, n,$$

where $\beta_j$ are the unknown coefficients that have to be estimated.

A continuous variable $x$ can be replaced by a differentiable monotonic function $g(x)$, without destroying the linearity of the model. Linearity is equally preserved if we take the product of two or more variables. Variables can also take the form of a factor with discrete levels. Let $x_j$ be a categorical variable with $k$ levels. $x_j$ can be replaced by dummy variables and written as

$$x_{j_1}u_1 + x_{j_2}u_2 + \cdots + x_{j_k}u_k,$$

where $u_i$ takes the value 1 if $x_j$ is at level i and zero otherwise. The maximum-likelihood estimates of the parameters $\beta$ are then obtained via ordinary least squares.

The SCAN*PRO model developed by Wittink et al. (1988) is an example of a linear model, and is one of the most widely used across the retail sales forecasting industry. SCAN*PRO is based on the premise that three factors influence sales: **1)** the price index[1], **2)** store display[1] and **3)** marketing campaign[1]. Its mathematical equation takes the form

$$S_{ist} = \lambda_{is} \cdot \mu_{it} \cdot \prod_{j=1}^{SKUs} \left\{ PI_{jst}^{\beta_{ij}} \cdot \gamma_{a_{ij}}^{PROMO_{jst}} \cdot \gamma_{b_{ij}}^{DISP_{jst}} \cdot \gamma_{c_{ij}}^{PROMO\&DISP_{jst}} \right\} \cdot e_{ist},$$

where:

$S_{ist}$ = sales of $i$ in store $s$ and week $t$

$\lambda_{is}$ = store-SKU factor for $i$ in store $s$

$\mu_{it}$ = week-SKU factor for $i$ in week $t$

$PI_{jst}$ = Price Index (price/average regular price) of $j$ in store $s$ and week $t$

$\beta_{ij}$ = Price elasticity between $i$ and $j$ (when $i = j$ it is the product's own elasticity)

$PROMO_{jst}$, $DISP_{jst}$, $PROMO\&DISP_{jst}$ = dummy variables for promotional campaign, display, or both, respectively

$\gamma_{ij}$ = multiplying factors of the promotional dummy variables

$e_{ist}$ = error component asociated to $i$ in store $s$, week $t$

[1] Both of the SKU being predicted and of the SKUs deemed to have a significant interaction with it. This way, not only the elasticities of the SKU are incorporated, but also cross elasticities.

### 2.2.4 Base-times-Lift Approaches

Some authors choose to define promotional sales as a multiple of baseline sales. According to Cooper et al. (1999), the lift factor is determined at a level of high aggregation, due to the limited amount of promotions for a specific item. Donselaar et al. (2016) measure the success of a promotion by the Lift Factor, defined as the sales during a promotion divided by the baseline sales. Baseline sales were measured by taking the average weekly sales of the non-promotional weeks during the five weeks preceding a promotion. The authors then use the natural logarithm of a Lift Factor $\ln(F_{lift})$ as the dependent variable.

A major advantage of using relative sales rather than absolute sales as the basis for the dependent variable is the fact that this results in standardized values for all promotions, making comparisons between promotions for different products more meaningful and easier.

### 2.2.5 Random Forests

The Random Forests algorithm is a refined version of the Bootstrap Aggregated (Bagged) Trees algorithm, which is itself an evolution of the more ordinary Classification and Regression Tree (CART) algorithm, systematized by Breiman et al. (1984).

Regression Trees successively split (branch) a dataset to maximize the homogeneity or "purity" of each "leaf". It does so using a greedy approach, splitting according to the feature that, at each splitting point (or "node"), explains the most variation in the training data. The prediction for new instances of data will inevitably be the value or average of the values of the response variable in the leaf correspondent to the features the new instances possess.

Tree predictors are excellent at capturing interaction effects between features since each leaf is formed according to a series of splits that are dependent on several features. However, being a non-parametric test, trees do not fare well at capturing relations between features and the response variable that could have been better described via a continuous function, be it linear, exponential, quadratic or logarithmic. Moreover, if grown without any limitations, trees tend to capture noise in the training data and overfit. As a result, either stopping criteria are defined for this algorithm - such as a minimum of observations at each given leaf, a maximum number of splits the tree can perform, or a level of purity that the algorithm considers to be enough and stop splitting through that branch - or manual "pruning" is performed to increase robustness.

To circumvent this limitation, Breiman (1996) proposes an ensemble approach where instead of a single tree being grown out of all the data, several trees are grown over different subsamples, which are random vectors sampled independently and with the same distribution for all trees in the "forest". When presented with new data, the results of all the trees will be averaged. This technique is called Bootstrap Aggregating, or Bagging for short.

Breiman (2001) further conjectures that the generalization error of a forest of tree regressors depends on the strength of the individual trees in the forest and the correlation between them. In

order to reap benefits from optimizing the Bias/Variance[1] balance, the author further refines the bagged tree algorithm, introducing randomness in the tree construction procedure. In the Random Forests algorithm, each node is only allowed to split over a random subset of features, instead of all of them. As a consequence, individual trees will be weaker predictors, however less correlated with all the other trees, increasing the robustness and predictive power of the algorithm.

### 2.2.6 Gradient Boosting Machine

A Gradient Boosting Machine is a similar algorithm to Random Forests, since it is also an ensemble of classification or regression trees. Friedman (2002) defines gradient boosting as additive regression models that sequentially fit a simple parametrized function (base learner) to current "pseudo"-residuals by least squares at each iteration. The pseudo-residuals are the gradient of the loss function being minimized, with respect to the model values at each training data point evaluated at the current step.

A more down-to-earth explanation could be: A Gradient Boosting Machine sequentially builds models that correct the mistakes that the previous ones made. While Random Forests grows trees in parallel, Gradient Boosting builds simpler trees (sometimes even decision "stumps", a tree with a single split) and stacks them.

Friedman (2002) shows that, similarly to the Random Forests algorithm, both the approximation accuracy and execution speed of gradient boosting can be substantially improved by incorporating randomization into the procedure. Specifically, at each iteration, a subsample of the training data is drawn at random (without replacement) from the full training data set. This randomly selected subsample is then used in place of the full sample to fit the base learner and compute the model update for the current iteration. This randomized approach also increases robustness against overcapacity of the base learner.

While being an algorithm that won many machine learning competitions, a critique made to this algorithm is that, in contrast to Random Forests, it will capture noise and overfit if too many models are stacked.

[1] According to Geman et al. (1992), the two components of forecasting error, correspondent to incorrectly modelling the data and capturing noise, respectively.

### 2.2.7 Other machine learning concepts

#### 2.2.7.1 Cross-validation

In order to compare models, it is necessary to have a reliable estimate of their predictive accuracy. The accuracy of an estimator is the average degree of error with which it predicts randomly selected instances, where we assume the distribution over the instance space is the same as the distribution that was used to select instances for the algorithm's training set.

Given a finite dataset, we would like to estimate the future performance of a model created by the given inducer and dataset. Kohavi (1995) summarizes the most common estimation methods and their comparative advantages.

The simplest is the holdout method, or test sample estimation, which partitions the data into two mutually exclusive subsets. The algorithm is fed a training set and the obtained model predicts the response variable on the test set. The resulting accuracy is a pessimistic estimator, since only a fraction of the available information is ever shown to the algorithm. This method poses a trade-off between the bias of the estimation and the confidence we can have on the accuracy estimated, depending on the size of the test set. This method has further disadvantages if the assumption that the data in the test subset originates from the same population than the training subset cannot be made, which is a serious possibility if the split is made according to a data feature (time, for example).

Cross-validation, typically designated k-fold cross-validation, splits a dataset into k mutually exclusive subsets of an approximately equal size. K models are produced and each is tested on the fold that was excluded on the respective training set. By averaging the accuracy of the various models, we can obtain a more reliable estimate than the holdout method. This method avoids biased estimates since all the data is considered both in the training and testing set. The confidence on the provided estimate increases with k, at the expense of more computational effort.

#### 2.2.7.2 Regularization

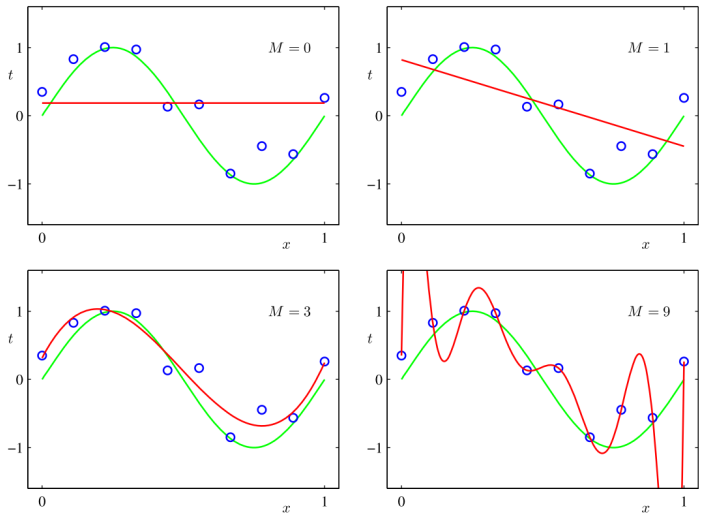
Considering the typical regression situation: we have a set of features and a response variable to be predicted. Unless instructed otherwise, a model will fit as best as possible to the training data. However, not all of the variation in the response variable can be explained by the input variables or at all. Overfitting will happen when a model is produced that corresponds too closely or exactly to a particular set of data, and may therefore fail to fit additional data or predict future observations reliably. An example of overfitting is displayed in the fourth pane of Figure 2.2.

In supervised learning settings with many input features, overfitting is usually a potential problem unless there is ample training data. Moreover, having a simple, interpretable model might be a criterion when performing regression. Thus, unless the training set size is large relative to the dimension of the input, some mechanism is needed to prevent over-complexity of the model.

One technique that is often used to control the over-fitting phenomenon in such cases is regularization, which involves adding a penalty term to a conventional error function (square loss, for example) in order to discourage model complexity. This leads to a modified error function of the form

$$\min_f \sum_{i=1}^{n} L(f(x_i), y_i) \; + \; \lambda R(f), \tag{2.1}$$

where $L$ is the underlying cost function that computes the cost of predicting $f(x)$ when the response variable takes the value $y$. $R(f)$ imposes a penalty on the complexity of the model $f$. The coefficient $\lambda$ governs the relative importance of the regularization term.




The base truth relating $x$ to $t$ is the green curve. Training data points, in blue, are obtained by adding random noise to the base truth. Even though the last model fits perfectly to the training data, a lower order polynomial best models the phenomenon. The first two models are *underfit*. Source: Bishop (2006)

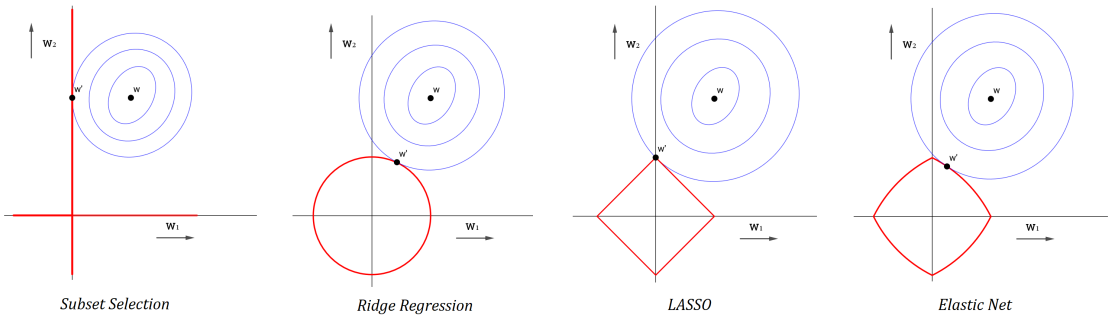
Figure 2.2: Plots of 4 models obtained via fitting polynomials of different orders M

Tibshirani (1996) describes two existing regularization methods, the subset selection method and the $L_2$ regularization method. Additionally, a new one is proposed, labelled as the Least Absolute Shrinkage Selecion Operator, or "LASSO", now also known as $L_1$ regression. Subset selection merely selects a predefined number of regressors in a greedy manner, or uses a scoring metric, such as the Akaike Information Criterion. The second, $L_2$ regularization, also known as Ridge Regression, uses a penalty term which encourages the euclidean norm of the parameters to be small. Both techniques have drawbacks - subset selection is highly sensitive to training data, and may exclude significant regressors from the model; Ridge Regression is more stable in this sense, but has no incentive to dropping coefficients. The $L_1$ penalizing term depends linearly on the absolute sum of the coefficients, combining the benefits of the previous two methods.

In an attempt to further exploit the potential of combining regularization methods, Zou and Hastie (2005) proposed the concept of the elastic net, where the $L_1$ and $L_2$ terms are used simultaneously, weighed using a parameter $\alpha$. A geometrical interpretation of these methods can be

visualized in Figure 2.3.

Regularization helps fulfil, as Domingos (1999) elegantly puts it, the role of Occam's razor in knowledge discovery - particularly the second razor mentioned in his paper, that postulates: "Given two models with the same training-set error, the simpler one should be preferred because it is likely to have lower generalization error."




The penalization isolines of the coefficients term and regularization term in blue and red, respectively, given a model $w$ with 2 coefficients, $w_1$ and $w_2$. Subset selection provides no incentive for reducing the coefficients' size, and Ridge Regression no incentive for removing them from the model. LASSO and the Elastic Net are compromises in between.

Figure 2.3: A visual interpretation of regularization methods

#### 2.2.7.3 Concept Drift

In machine learning, the supervised learning problem is formally defined as follows: In regression tasks, we aim to predict a target variable $y \in \Re^1$ given a set of input features $X \in \Re^p$. Because data is expected to evolve over time, its underlying distribution can change dynamically. Moreover, the underlying phenomenons and rules that map the feature space $X$ to the response variable $y$ may also change over time. The latter is referred to as *Real Concept Drift*, while the former as *Virtual Concept Drift*. Gama et al. (2014) formally define concept drift between time points $t_0$ and $t_1$ as

$$\exists X : p_{t_0}(X,y) \neq p_{t_1}(X,y).$$

In the light of (promotional) retail sales, real concept drift takes place when preferences and behaviours of customers change. Changes in the retailer company's policies may cause either virtual (i) or real (ii) concept drift, depending on how these changes affect consumer behaviour. For instance, if the company chooses to decrease their average discount rate, but promote on average more products at the same time (a change in $X$), we expect customers to buy differently (a change in $y$), but not because their preferences have changed, rather they are responding to their internalized price elasticities, which have been accounted for. Rather, if the company decides, *ceteris paribus*, to increase promotion frequency, to a rate equal or lower than the inter-purchase time, customers might be "trained" to frequent the stores only in promotional periods. Non-promotional weeks will suffer as a consequence, denouncing a change in consumer behaviour - $y$ has changed, despite a constant $X$.

In the retail context, concept drift, if existent, is assumed to follow an incremental or gradual pattern.

Techniques to circumvent gradual concept drift include:

- Frequently updating the model with recent data;
- Establish a "rolling window" period so the model forgets the oldest data;
- Weighing data according to its age;
- Learning the change, where a subsequent model trains on new data and corrects the predictions of precedent ones.

#### 2.2.7.4 Hyperparameter tuning

We defined above what the objective of a typical machine learning algorithm is. In most cases, learning algorithms have a set of levers $\theta$ that control the inner workings of the algorithm, designated *hyperparameters*. So far we have mentioned the regularization parameters $\lambda$ and $\alpha$. The Random Forests algorithm has several, such as the number and maximum depth of trees or other stopping criterion. In practice, it is necessary to select these hyperparameters so as to minimize generalisation error. This problem is referred to as hyperparameter optimization.

In reality, there are no efficient optimization methods for performing such task, which leads practitioners to prefer a trial search instead, according to Bergstra and Bengio (2012).

A very common method is grid search, where $\prod_{i=1}^{n} l_i$ models are built for $n$ hyperparameters and $l$ levels per parameter. Bergstra and Bengio (2012) suggest a slightly different search method, where the levels searched on are random, meaning that each parameter is searched in far more levels than with grid search. This method proves to be superior because the generalization error tends to depend on only a few of the $n$ hyperparameters, but there might be global maxima otherwise missed by the grid search.

A different approach is the one proposed by Laan et al. (2007) where an ensemble of models is created from a grid search and then a weighing meta-algorithm combines the results of each according to cross-validation results.

## 2.3 Factors that influence demand

This section enumerates and explores the major factors considered by academia to influence baseline and promotional sales.

### 2.3.1 Time series and events

Many product lines are affected by seasonality, typically following a yearly cycle. Seasonality in demand may be caused by seasonality in the availability of the goods, as is the case of fruits, where some species are very dependent on weather and are only harvested at certain periods of the year. It can also be due to events or periods to which the consumption of certain products is associated, like Christmas or Easter. Lastly, weather variations throughout the year may have

implications in the adequacy or attractiveness of certain products (e.g. barbecues in Summer, hot chocolate in Winter). Modifications to autoregressive models can be done to account for seasonality, such as the SARIMAX method used by Aburto and Weber (2007). It is important to understand that demand may not only cycle through the year, but also month (e.g. effect of salaries) and week (weekdays and weekends), as incorporated in the models of Arunraj and Ahrens (2015).

Specific events, regardless of where they fall under the yearly, monthly, or weekly cycle, are also usually modelled for. Aburto and Weber (2007) and Arunraj and Ahrens (2015) have shown these factors to have significant impact in forecast accuracy:

- National or local holidays, and the day before or after;
- School vacations;
- Festive periods, or specific variables for Christmas, Easter, and weeks before or after them;
- Festivals and events, to account for temporary population mass transfers;
- Starting and ending days for promotions.

Nearly all of relevant literature includes time series and/or events in their models. It is advised, though, to test the variables for spurious correlation and cross-validate the models for overfitting.

### 2.3.2 Store

Several authors include in their approach the characteristics of the stores. Stores in touristic locations, for instance, have a higher percentage of occasional customers than a countryside store. Moreover, the target demographic of a store also plays a role on how demand behaves. Arunraj and Ahrens (2015) include the ratio of irregular customers in their models, while Hoch et al. (1995) includes variables such as: median income, percentage of women who work, level of education, age distribution, average household size, and real estate valuation.

Ailawadi et al. (2006) have shown that whereas average sales increase with store size, larger stores tend to have less promotion profitability.

### 2.3.3 Product

The characteristics of the product and the way it presents itself to the customer are also a variable deemed as relevant by several authors. The quality of a delivery, even if controlled by the retailer, depends mostly on the manufacturer. It is usually presumed that freshness and good physical condition of a product positively affect its sales. It is not trivial to translate these characteristics into a forecasting model. In the work of Arunraj and Ahrens (2015), shelf-life and package size have been used as proxies for these properties.

The product shelf-life affects yet another pattern in consumer behaviour - *stockpiling*. Marketing practitioners use this term to refer to the pre and post-promotional effect of sales reduction. Promotions encourage the customer to buy more during such periods and maximize savings. Since promotions are often communicated in advance, customers tend to delay their purchases and acquire the desired product at a discount, an effect known as *pantry-loading*. Narasimhan et al. (1996) found a significant relation between the ability to stockpile and the promotion bump.

Thirdly, shelf-life in combination with frequency or rate of consumption influence the purchase cycle of products, also known as inter-purchase time. Narasimhan et al. (1996) have further discovered that an increase in this time influences negatively the price elasticity of a product's sales. The reason for this is that customers are less willing to switch to a less-preferred brand since they would have to live with that decision for longer. It also discourages pantry-loading because the product has to be stored for a longer period of time.

Lastly, and probably the most important property of a product, a product's base retail price and its *price elasticity* greatly influence sales. It figures in this subsection, even though the characteristics of the product alone are far from being the only ones that influence price elasticity, as shown by Hoch et al. (1995). Price is the figure to which the customer compares a product's utility, and is seen as the major sales driver. Price elasticity ($\varepsilon_p$) is represented by the quotient between demand variation and price variation, as shown in Equation 2.2:

$$\varepsilon_p = \frac{\%\Delta\ quantity}{\%\Delta\ price} = \frac{\frac{\Delta\ quantity}{quantity}}{\frac{\Delta\ price}{price}} \tag{2.2}$$

Price elasticity has, with very few exceptions, a negative signal, meaning that a decrease in price will almost always lead to an increase in demand. It is important to note that:

- If $\varepsilon_p < -1$, a decrease in current price will increase both sales quantity and revenue.
- If, however, $-1 < \varepsilon_p < 0$, a decrease in price will increase unit sales, but decrease revenue.

Price elasticity is, therefore, taken deeply into account by retailers when deciding on their assortment and base retail pricing. It is also a determinant factor of promotion profitability, even though the price cut is not the only driver of sales during promotions.[2]

### 2.3.4 Promotions

Promotions represent nowadays, as we have seen, a significant portion of grocery retail sales.

The retailer might use several mechanisms to promote its products, and these are, among others: **a)** Percentage discount **b)** Direct discount in € **c)** "Buy X get Y free" **d)** "X for €Y" **e)** Cumulative discounts in loyalty campaigns **f)** Bundle deals and **g)** Discounts in partner businesses. Despite the fact that, with greater or lesser ease, these discount formats can be converted into a relative price cut, some authors choose to include in their models, as a flag indicator, the promotion type, as seen in the work of Ramanathan and Muyldermans (2011).

The price cut mentioned above can be understood as the deal intensity, and is used with rare exceptions in all promotional sales forecasting models. However, the way it is incorporated is a subject of divergence, since whereas authors such as Narasimhan et al. (1996) or Ali et al. (2009)

[2] In fact, promotions are usually accompanied by quantity discounts by the manufacturers, improving their profitability from the cost side. Promotions may also drive more customer traffic, increasing sales in general, or increase sales of complementary products. Many promotions are also a reaction to competition, to avoid decrease in customer traffic. Finally, there is the case of promotions made to drain excess inventory which would spoil otherwise.

use the price cut in cents as a variable, Ramanathan and Muyldermans (2011), Ailawadi et al. (2006) and Ma et al. (2016) use the relative discount.

In order to maximize consumer awareness about the existence of a promotion, price cuts are in almost all cases accompanied by some form of marketing campaign. Communication channels include newspapers, radio, TV, SMS, and the retailer's own promotional material, which can be sent via mail to the customer and is usually available at the entrance of the store. The detail of integration of the campaign's properties in forecasting varies. Ailawadi et al. (2006) merely include a flag indicating whether a promotion is featured, whereas Kuo (2001) includes a flag for each deal support format, and Koottatep and Li (2006) differentiate features in front-, middle- and back-pages of promotional material.

Inside the stores, another array of promotion techniques is used to draw the customer's attention to promotional items, from the common promotional tags, to islands or aisle top displays. Similarly to awareness campaigns, this information is used at different levels of detail in forecasting models.

## 2.4 Product interaction

A model built only using the factors mentioned in the previous section would consider a product as an isolated element within the store's assortment. However, customers often enter a store with a set of needs that can be satisfied by a non-unique set of products. This last statement can lead us to infer that the sales of a product are not only influenced by its own selling conditions, but also by other products'.

An economic concept that can be used to describe how the selling price of a product affects the sales of another is the cross elasticity of demand, and is calculated by taking the percentage change in the quantity demanded of one good and dividing it by the percentage change in price of the other good.

$$\varepsilon_{p_{i\rightarrow j}} = \frac{\%\Delta\ quantity_j}{\%\Delta\ price_i} = \frac{\frac{\Delta\ quantity_j}{quantity_j}}{\frac{\Delta\ price_i}{price_i}} \tag{2.3}$$

Notice the notation used in Equation 2.3 ($i \rightarrow j$) which is not symmetric. The price of product $i$ might have influence in the sales of $j$, but not necessarily vice-versa, as shown by Donselaar et al. (2016).

Liu et al. (2016) developed a conceptual framework that includes the effects of promotions on products believed to be substitute and complementary. The results obtained support the conceptual framework, as increases in the price of the studied product lines (spaghetti) increases the sales of competing brands, but decreases the sales of a complementary product (spaghetti sauce).

Hruschka et al. (1999) approach the problem of dimensionality in finding pairs with significant interactions by assuming that items bought together more frequently are complementary, or substitute if the opposite occurs. Hruschka et al. (1999) then expand a multivariate logit model developed in an earlier paper with promotional information from other products.

Vindevogel et al. (2004) find significant pairs in a similar manner, but estimates product interactions via a Vector Autoregressive (VAR) model. Dawes (2012) aggregates sales at the category level and adds inter-category promotional variables to the SCAN*PRO model.

Ma et al. (2016) conduct Granger causality tests to identify promotional interactions and use LASSO to reduce the number of SKUs considered to be influential. To integrate promotional information of other variables, Ma et al. (2016) use a model with intra-category promotional information to predict the residuals of a first model developed using only the SKU's own predictors. A third model is used to further reduce residuals using inter-category information. The additional intra- and inter-category information was able to improve the accuracy of the first model by 12,6%, and 95% of the added value comes from intra-category information.

# Chapter 3

# Problem description

In the first section of this chapter, the forecasting and replenishment methodologies currently in use at the retailer's Fresh Division will be described in detail. In the second section, both practical and theoretical limitations of the current methods will be discussed.

The objective of the project is, since the beginning, to improve the forecasting and replenishment procedures at the retailer's Fresh Division. Before the project's start, the forecasting procedure for promotional sales resumed itself to selecting the most similar historical campaign, based on week of the year and selling price. There was no rigorous method to measure "closeness" between historical campaigns and the one to be forecast. Moreover, this procedure was undertook manually for every single promotional item, making it time-consuming and prone to error. Variables such as the vehicles of communication used or seasonal events were accounted for subjectively, and the occurrence of product shortages in past campaigns was not considered at all.

In the initial phase of the project, before this thesis, the methodology developed by Batista (2016) was implemented. This methodology comprises of collecting and processing of the retailer's historical data, a causal forecasting algorithm, and two applications designed to forecast sales and generate replenishment parameters, respectively.

## 3.1 Current methodology

### 3.1.1 Available Data

The data collected concerns five aspects of the retailer's business:

**Sales** The retailer's information systems collect and store end-of-day sales at the SKU-location level. Specifically, the sales volume in units and Euro, price, and reported net sales (sales absent tax) are collected.

**Store information** Stores are organised in a hierarchical structure, aggregated by format and geographical distribution, as shown in Figure 1.4. Information regarding the store's assortment is necessary for forecasting, and each store's delivery windows condition replenishment parameters. Moreover, some stores do not have counters (butcher's, fish, take-away) and therefore entire product categories are not available at certain stores.

**Promotional information** Information from promotional campaigns is available at the SKU/-Day/Store Format level. Each entry contains information about the promotion's properties, such as the percentage discount, communication vehicle, and whether the discount is direct or accruable in a customer card. Percentage discount ranges from 0 to 50 percent, and has the distribution visible in the left graph of Figure 3.1. Promotions are publicized using the means listed below. Their relative importance can be seen in the pie chart in Figure 3.1.

**Weekly Brochure** A brochure sent by post or email concerning a full week of promotions. It is the main promotional vehicle of the company;

**Leaflet** A one-paged leaflet concerning 2 to 4 days, containing less products, available only in-store;

**Television Ads** TV commercials, which have a variable duration and decided product-by-product, and

**Thematic** Themed leaflets, where a specific assortment of products is promoted with low discounts for a longer period of time (2 to 4 weeks).

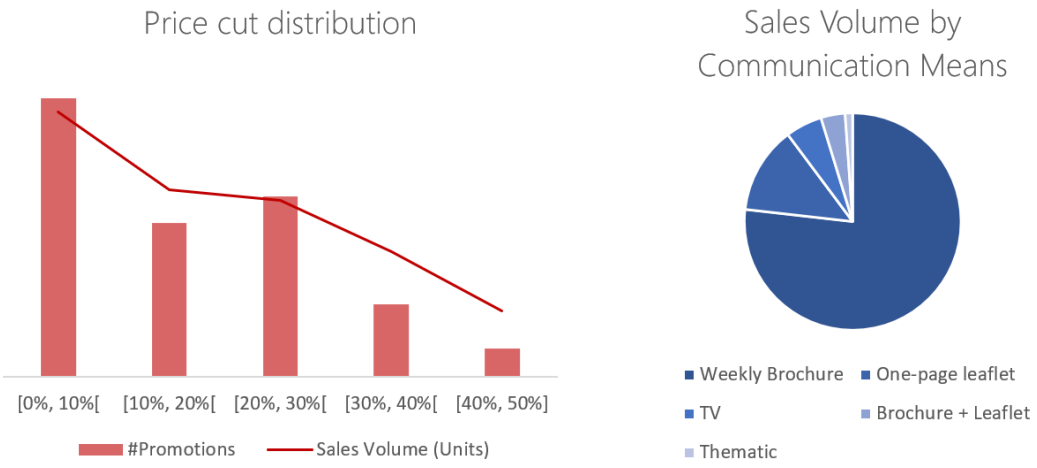



Figure 3.1: Price cut and Communication Means relative importance

**Stocks** Daily stocks are automatically accounted for by adding the previous day's stock and the difference between deliveries and sales. This system is not fail-proof, since products might get spoiled, or stolen, and deliveries and returns might be poorly registered. Thus, store managers validate the store-level stock periodically and correct the entries manually.

**Product information** Products are also organised hierarchically in a product structure, already outlined in Figure 1.3. Calculations are done at several levels of aggregation according

to this structure. For the replenishment module, additional product characteristics are imported, such as shelf-life, ship-pack size, presentation stock, and supplier lead times.

### 3.1.2 Solution

The solution in place consists of several stages and components, and is depicted in Figure 3.2.

1. An ETL (Extract, Transform and Load) application that retrieves and preprocesses sales and promotional data from the retailer's Information System, consolidating them in a single table;
2. An R script that firstly adds and transforms features of the dataset, and then generates forecasting models at different levels of aggregation;
3. A forecasting module where the list of promotional items to be forecast and respective promotional characteristics is introduced, as well as the model coefficients; Aggregate predictions are made for the following week for each respective Store-Format, validated and then disaggregated at the Day/Store/SKU level;
4. The replenishment module that transforms sales predictions into replenishment parameters to be later introduced in the retailer's Retailer Merchandising System (RMS).

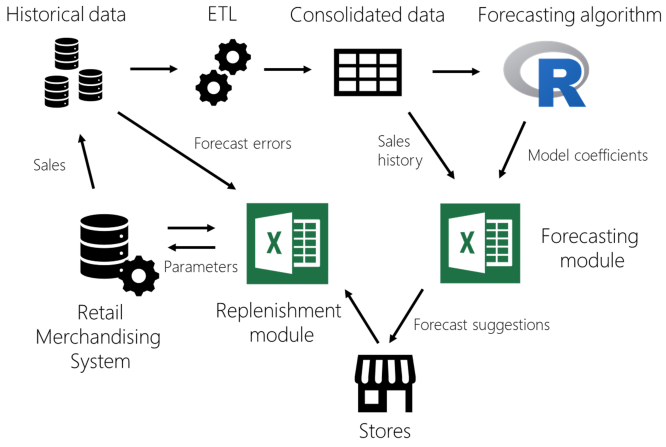



Figure 3.2: Forecasting and Replenishment solution schematics

An important issue with forecasting demand is that historical demand does not necessarily correspond to historical sales. The fact that, for instance, product "A" is unavailable at a certain store will force the customer to either not purchase or to purchase an alternative product "B". In this case, demand for product A was not converted into sales of product A, and was transferred, depending on the willingness of the customer to substitute, into sales of product B.

Since the retailer is interested in forecasting demand, not sales, but only historical sales are available, demand has to be estimated in cases that product availability was compromised. Thus, sales are confronted with the day's stock in order to identify shortages. With the purpose of estimating what the store would sell if there was no shortage, an analysis of the typical weight

of each store in the respective store format is conducted. Whenever a store's stock of a certain SKU goes to less than 20% of the sales of the next day, it is assumed that there was a product shortage. This is a conservative estimate, since it is necessary to account for wrong stock levels in the information system, reduced sales due to insufficient presentation stock, and stock that might be kept at the store's back-end. The SKU's sales of that day are corrected to correspond to the typical proportion of sales that store represents. A typical case of a weekend shortage is depicted in Figure 3.3.

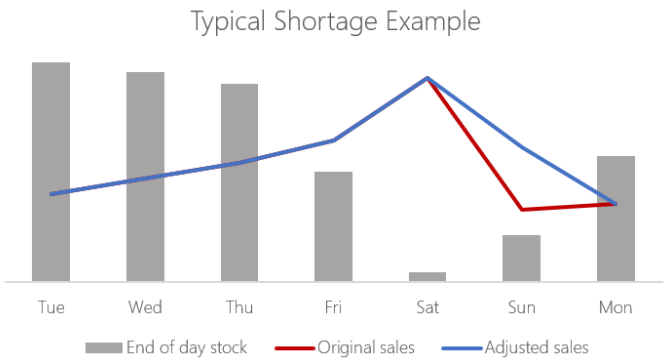



Figure 3.3: An example of sales correction given a product shortage

The adjusted sales are joined with the promotional information database. The price "index" is obtained by first obtaining the sales price, by dividing gross sales by unit sales, and then dividing the price by the regular selling price of the trimester. Promotional information is added, and the resulting summary table contains data since the beginning of 2014 and is updated monthly with new data.

### 3.1.3 Variables used

The business variables currently considered to explain promotional sales are the following:

**Baseline Sales** This variable estimates what the sales would be/have been if no promotions or stock-outs took place. It is calculated using a 5-week moving average.

**Price Index** A proxy for promotional intensity, the Price Index represents how the selling price relates to the average regular selling price. Since the retailer has a maximum discount of 50%, the Price Index is therefore always between 1 and 0.5. As we have seen in Section 2.3.3, price is one of the major driver of sales.

**Weekdays** The weekday cycle, as exemplified in Figure 3.3, is very notorious. The variability respective to the days of the week is captured with 7 flag variables.

**Communication Vehicle** The way the public gets to know about the promotions is also deemed to be relevant. This business aspect is included via 4 flag variables, one for each communication means.

**First days** It was determined that, for promotions other than the weekly brochure, the first day of promotion had a sales boost.

**Promotional quota** The percentage of items, weighed by their sales volume, that are being promoted at any given moment in the same subcategory. This variable is intended to measure the promotional intensity of articles similar to the product being studied.

**Festivities** As shown in Section 2.3.1, some articles have particular sales patterns in festive periods. Specifically, the festivities considered are Christmas and Easter.

**Seasonality bumps** Some articles only sell in specific periods of the year. For every week of the year, its sales are compared with sales of the previous and following 5 weeks, and the model identifies a seasonality bump if the least selling week of the following 5 has 5 times as much sales volume as the highest selling week of the previous 5. A reciprocal logic is used to identify exits of seasonal periods.

**Store quota** The number of stores, weighed by their sales volume, that carry a given product in their assortment. This allows the model to adjust its predictions when the number of stores carrying that product in their assortment changes.

**Seasonal Index** The ratio between the average sales of a given week of the year compared to the yearly average. This index is not used explicitly as a variable, but changes the baseline forecast to include seasonal effects.

Naturally, the variables in the test dataset cannot be estimated using the data of the present or future. Therefore, seasonality entries, exits, and seasonal indexes are assumed to have an equal profile every year. Baseline sales use a weighed rolling average of only the previous 3 regular weeks. The seasonal index is used to correct baseline sales to account for seasonality.

### 3.1.4 Model formulation

In order to overcome the limitations of the previous methodology used to forecast sales, the possibility of describing promotional sales as a consequence of a set of variables was studied by Batista (2016). Such an approach would derive the response of sales to the conditions the sale is made in, many of which are known *a priori*, and eliminate the need of manually selecting a historical campaign every time a new promotional campaign is to be forecast. Even though the period of prediction is usually a week, given the prevalence of the weekly promotions as shown in Figure 3.1, the model still has to be able to place daily forecasts, since the other kinds of promotions last less or more than a week.

Several models were studied and the one chosen is a derivative of the SCAN*PRO model developed by Wittink et al. (1988). The model was adapted to the retailer's available data and dimension. For instance, the interaction between products was not considered as it would exponentially increase the number of coefficients of the model. Variables are iteratively added and

removed from the regression in order to produce the best performing model. Batista (2016) models sales at the SKU/Store Format/Day level using the formula:

$$\ln(S_{ist}) = \alpha_{is} \cdot \ln(PI_{ist}) + \beta_{is} \cdot \ln(B_{ist}) + \sum_{w=1}^{7} \gamma_{1isw} \cdot W_{tp} + \sum_{m=1}^{4} \gamma_{2ism} \cdot M_{istm} + \gamma_{3is} \cdot F_{ist}$$
$$+ \gamma_{4is} \cdot SC_{ist} + \sum_{h \in \{C,\, E\}} \gamma_{5ish} \cdot H_{th} + \sum_{e \in \{in,out\}} \gamma_{6ise} \cdot E_{ite} + \gamma_{7ist} \cdot N_{ist} + \sum_{d \in \{D,\, C\}} \gamma_{8isd} \cdot D_{istd} + \varepsilon_{ist}$$

where:

$S_{ist}$ = sales of $i$ in store format $s$ and day $t$
$PI_{ist}$ = Price Index (price/average regular price) of $i$ in store format $s$ and day $t$
$B_{ist}$ = Baseline sales estimation of $i$ in store format $s$ and day $t$
$W_{tw}$ = Dummy variable that takes the value 1 when weekday($t$) = $w$
$M_{istm}$ = Dummy variable assigned 1 when $i$ is communicated via $m$[1] in format $s$, day $t$
$F_{ist}$ = Dummy variable that takes the value 1 when $i$ is in its first promotional day in store format $s$ and day $t$
$SC_{ist}$ = Promotional quota of $i$'s subcategory in store format $s$ and day $t$
$H_{th}$ = Dummy variable that takes the value 1 when $t$ is in holiday period $h$[2]
$E_{ite}$ = Dummy variable that takes the value 1 when $t$ corresponds to either an entry or exit of seasonality of $i$
$N_{ist}$ = Weighed percentage of stores selling $i$ in store format $s$ and day $t$
$D_{istd}$ = Dummy variable that takes the value 1 when the promotion is type $d$[3]
$\alpha_{is}, \beta_{is}, \gamma_{is}$ = model coefficients
$e_{ist}$ = error component associated to $i$ in store format $s$, day $t$

The coefficients are estimated via a stepwise regression that contains at least $\alpha$ and $\beta$ and greedily adds and removes coefficients in order to produce the best score. The score is calculated via Akaike (1974)'s Information Criterion (AIC), which is defined as

$$AIC = 2k - 2\log(\hat{L}),$$

where $k$ is the number of coefficients, and $\hat{L}$ is the maximum value of the likelihood function for the model. As mentioned in Chapter 2, AIC depends on the number of coefficients, being part of the *Subset Selection* metric family.

This model is run for every SKU, if it belongs to sales class A[4] of a certain category and has at least 120 observations, and for every base unit. According to Batista (2016), this assumption produces more robust models for slower moving SKUs and new products with little historical data. Moreover, observations that distance more than a year from the training period's end count half as much to the loss function, and promotional observations count five times as much as a regular

[1] Weekly brochure, leaflet, TV or themed leaflet
[2] $C$ for Christmas, $E$ for Easter
[3] $C$ for Card, $D$ for Direct Discount
[4] According to Pareto's rule, class A products collectively represent 80% of the sales volume.

sale. The coefficients and seasonal indexes obtained are exported, to be imported later by the forecasting module.

### 3.1.5 Forecasting and replenishment procedure

The superior forecasting methodology had to be translated into a new *modus operandi* at the retailer. In order for the improved accuracy of the model to carry on to a better performance at the supply chain, a solution that allowed stock managers to use the model to forecast sales and validate the model's predictions, as well as to translate predictions into orders to be delivered at the stores, was necessary. A framework that comprises a cycle of procedures was developed, revolving around each promotional week, as well as two interfaces aimed at forecasting sales and generating replenishment orders. The mentioned cycle is outlined in Figure 3.4 and explained below.

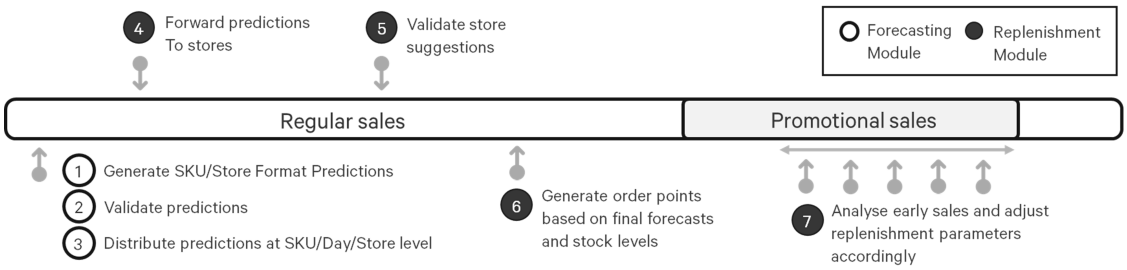



Figure 3.4: Forecasting and Replenishment procedure for a specific promotional week

1. In the forecasting module:

   (a) The forecasting module imports the coefficients and seasonal indexes generated by the forecasting algorithm. Stock managers input the promotional assortment and characteristics to be forecast. The module estimates baseline sales from previous weeks without promotions.

   (b) Forecasts are generated at the SKU/Store Format/Day level. However, throughout the initial phase of the project it was found that aggregating the weekly forecast and redistributing it through the week via weekday indexes was more stable and accurate.

   (c) A validation dashboard allows stock managers to validate aggregate forecasts by comparing them with the most similar historical campaign and looking at past sales.

   (d) After validation, the module distributes aggregate forecasts by each store. These results are then exported and sent to the stores, that may suggest changes.

2. In the replenishment module:

   (a) Store suggestions are imported and accepted/declined by the stock manager. These final forecasts are exported to suppliers a week before the promotional week.

   (b) Supply chain parameters, such as supplier lead times and delivery days, store delivery windows, product ship-pack sizes and flow type (PBL[5] or PBS[6]), are combined with the forecasts to generate cycle stock quantities. Since there is some error in the predictions, an additional quantity of safety stock is considered according to the intended service level and historical forecast error. The replenishment method is the (R, S) method[7], and therefore the results are inventory levels to be met each day. These inventory levels are exported to the retailer's ERP, that places orders each day.

## 3.2 Limitations

The implementation of the methodology described in the previous section shed light over some inherent drawbacks:

**Overfitting** The algorithm was developed and optimized using data of a single product category, which is also object of study of this thesis - the Fruits category. This category has very specific characteristics, as mentioned in Chapter 1, which have been very clearly accounted for in the forecasting algorithm. There is a very meticulous treatment of seasonality, which produced a good result in the Fruits category, but there is no guarantee it will not undermine the ability to produce accurate forecasts in other categories.

Moreover, there is no control over the sizes of the model's coefficients. Given the multiplicative nature of the model, the result is such that every other week, there is an SKU which predictions are completely off the scale, usually because one of the variables has gone off the range in which the model was trained on. This creates confusion among stock managers and discredits the methodology.

**Assumptions** Despite the fact that the model is being used in a promotional-only context, it is designed to predict both regular and promotional sales. This assumes that promotional and regular sales come from the same distribution and that the elasticities to variables such as holidays or subcategory promotional quota are the same in both cases.

In addition, the model contains variables that are not used later on to predict sales. For example, in the forecasting module, predictions are in fact generated at the SKU/Store Format/Day, but aggregated and redistributed throughout the weekdays and stores according to the typical weight of the week day in the product's category. There is variability being

[5]Pick By Line is a flow type with no stock at the warehouse, where suppliers' deliveries are shipped to stores in the same day.

[6]Pick By Store involves having a level of buffer stock at the warehouse, and is more suitable for slow-moving products with high shelf-life.

[7]This replenishment method is described in Appendix A.

explained by 7 coefficients that later on have no effect on the forecast, indicating potential for improvement.

Lastly, even if more recent observations have more weight, the model is still using data from the beginning of 2014. Since then, both company policies and the grocery retail industry have had changes. For instance, promotional activity has increased, as described in the previous chapters. Also, the company has shifted most of its promotions to the direct discount mode, slowly abandoning loyalty cards, except for occasional campaigns, as shown in Figure 3.5.

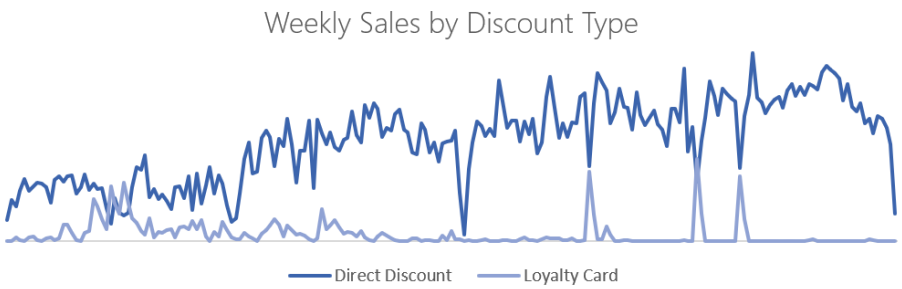



Figure 3.5: Weekly Sales by Discount Type

**Cannibalization** Including as a variable the subcategory promotional quota does, in fact, improve the model's accuracy. However, it is a rather superficial and indirect way of modelling product interaction, and calls for further investigation on this subject.

**Bad data** The fact that promotions are catalogued manually makes this information prone to mistakes. In practice, a non-negligible set of promotions never make it to the database and the data processing mechanism ends up marking promotional sales as regular. This has an especially negative impact when computing baseline sales, exaggerating the resulting predictions.

**Results** All the previous topics are reflected in the less than satisfactory results in the selected test period. The results of the model over roughly 5 months of sales in 2017 are depicted in Table 3.1. For comparison, the results of the model currently in use at the retailer to forecast baseline sales will be analysed further in Chapter 4, and are able to predict sales with half the forecasting error. Promotional sales are naturally harder to predict, but Table 4.1 provides a reference to aim at.

Table 3.1: Results of the current model on the selected test period

| Category | MAPE | Bias |
|---|---|---|
| Fruits | 30.5% | 1.1% |
| Chicken | 38.8% | 20.0% |
| Frozen Fish | 37.1% | 5.2% |

# Chapter 4

# Methodology

This is the chapter in which the methodologies used to overcome the identified setbacks and to improve the forecasting performance of promotional sales are described. After a summary description of the project timeline, a novel approach to data processing is detailed, the new model formulations are specified, and the methodologies for model selection and optimization, as well as the integration of product cannibalization, are characterized.

## 4.1 Approach

The timeline for this phase of the project at the retailer was defined internally to span the last four months of 2017, as portrayed in Figure 4.1. Initially, it was necessary to acquire the necessary knowledge and tools. The execution part of the project consists in putting into practice the previous learnings by translating the algorithms and procedures into code, making sure that reliable performance estimates were produced and that the optimization methodologies used would lead to satisfactory solutions. Lastly, the timings of the project allowed the topic of product interaction to be studied and included.

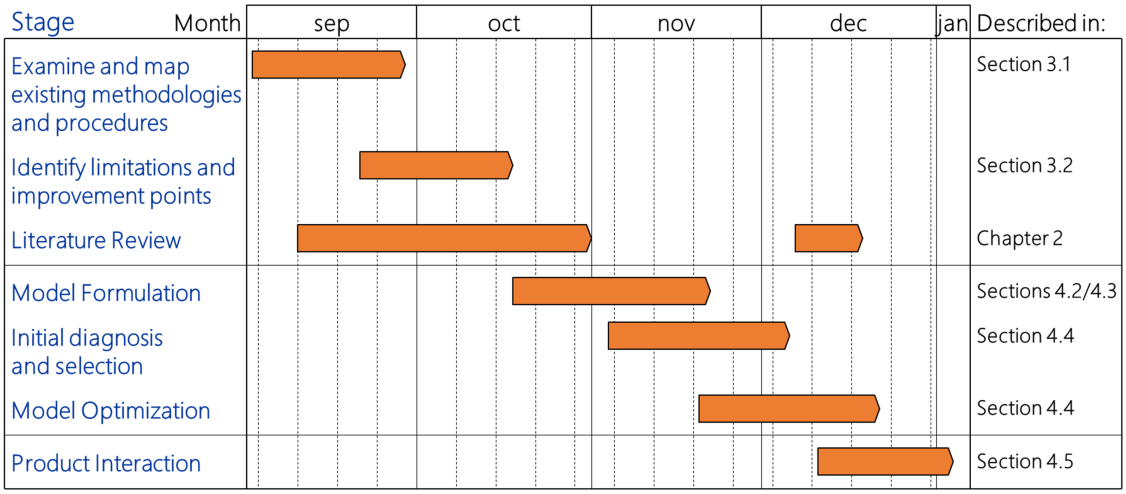



Figure 4.1: Illustrative timeline of the project

## 4.2 Inputs

### 4.2.1 Sales Baseline

The sales baseline is a fundamental input for promotional sales forecasting, since it is present in all models in relevant literature. Batista (2016) estimates the p-value for the significance of this variable to be 0,2%. In spite of its importance, one of the major drawbacks of the previous methodology was the complex calculation method of the sales baseline. Often, promotional sales would be labelled as regular due to the manual input of promotions, which significantly disturbs the baseline. Recently, the retailer adopted a commercial demand forecasting tool designed to forecast weekly regular sales. This forecasting system has 4 forecasting models available:

**a)** Simple Exponential Smoothing;
**b)** Trend Exponential Smoothing, both in its additive and multiplicative version;
**c)** Seasonal Exponential Smoothing, only available when there is at least a year of sales history;
**d)** Croston's Intermittent Model, indicated for slow movers.

The system begins by preprocessing sales data, removing outliers, promotional sales, and shortages. Afterwards, for each SKU-store pair, the algorithm aggregates data at all possible combinations of levels at the Store and Product structure. For each combination, the model evaluates every model from the list above and selects the best performing one at the respective SKU-Store according to internal metrics. This algorithm is ran weekly, which means that for the same SKU-Store pair, the aggregation level and model used can be different every week[1].

In order to take advantage of a superior and, more importantly, more stable sales baseline, the previous baseline computing method will be dropped. The sales baseline and baseline forecast will be retrieved directly from the forecasting tool. The baseline sales will be used to *train* the model, but in the testing period, the one-week baseline forecast will replace the baseline sales to provide a realistic estimation of the final model accuracy. This system's forecast has, naturally, a degree of error. Its MAPE and Bias, at the Week/SKU/Store Format level for the test period, can be seen in Table 4.1. It is expected that (some of) this error is reflected in the promotional forecasting error.

Table 4.1: MAPE and Bias of the baseline forecast in the test period

| Category | MAPE | Bias |
|---|---|---|
| Fruits | 16.8% | -0.5% |
| Chicken | 11.6% | -0.4% |
| Frozen Fish | 15.1% | -0.1% |

[1] In practice, only very few SKU-Store pairs change from one week to the other.

### 4.2.2 Variables

To improve coherence with the forecasting module, which distributes sales through the week using the historical weight of each weekday and ignores the elasticities of the model, it would be convenient to drop the flag indicators for each day of the week. However, predictions are still to be made for each day, since not all promotions last an entire week, and the weekday effect is very notorious. The solution found was to distribute the weekly baseline/forecast from the commercial forecasting system using the same logic as the forecasting module. This way, the weekday effect is still available for the model to consider, with the advantage of removing 7 variables from the model.

Moreover, besides the variables described in Section 3.1.3, there were new variables added to the dataset:

**Week of the month** Paydays and pension payments often fall in specific days of the month. To capture a potential monthly cycle, each day is assigned values 1, 2, 3 or 4 depending on which quarter of the month they fall in.

**Weekday Holidays** Whenever a holiday falls on a weekday, it is believed have an impact on the purchase patterns and quantities on that day, which are assigned the value 1, and the ones adjacent to it, assigned the value -1.

**Time since last promotion** To capture eventual pantry-loading effects, the time since the last promotion of a given SKU is classified in "Last Week", "Last Month", and "None".

### 4.2.3 Training and test datasets

Given our purpose of only forecasting promotional sales, the final dataset only includes promotional observations. The training period was defined to be at least a year, to capture all eventual seasonal effects. The retailer keeps daily stock information only since June 2016. As we have seen, this data is necessary for adjusting sales, and as a consequence the training period spanned from that date until June 2017. The test period spanned from the end of the training period until the beginning of November, 2017. The dataset contains 512 sales days, 29% of which inside the test period. The dataset size and distribution is represented in Table 4.2.

Table 4.2: Number of observations per category

| | Train Period | Test Period | Total |
|---|---|---|---|
| Chicken | 4701 | 1665 | 6366 |
| Fruit | 11175 | 5610 | 16785 |
| Frozen Fish | 26530 | 10513 | 37043 |

Even though the Frozen Fish category contains half as many SKUs as the Fruits category, the much larger dataset can be explained by both the higher promotional frequency and the general "unseasonality" that these products are subject to, therefore selling all year round.

## 4.3 Formulations

It is assumed that the variables selected contain information that can explain sales. Therefore, there must be a base phenomenon that maps these business variables to the sales volume. To forecast sales, the best performing model would be the one that most resembles the base phenomenon. Since this mapping function is unknown, it will be approximate it via inducers, or algorithms. It is important to have in mind the difference between these two concepts: **1)** An *algorithm* is the set of procedures that approximates a function that maps the independent variables to the dependent one, while a **2)** *model* is the approximate mapping function itself. Algorithms are usually named after the sort of model they produce. Four different algorithms were used:

**Regularized Linear Models** The developed linear regression is an extension of the one developed by Batista (2016). It differs in a few of the variables chosen and it is not obtained by stepwise regression, but instead by Elastic Net regression. Its formulation is as follows:

$$\begin{aligned}\ln(S_{ist}) \;=\; & \gamma_{1is}\cdot\ln(PI_{ist}) \;+\; \gamma_{2is}\cdot\ln(B_{ist}) \;+\; \gamma_{3is}\cdot SI_{ist} \;+\; \sum_{m=1}^{4}\gamma_{4ism}\cdot M_{istm} \\ & +\gamma_{5is}\cdot I_{ist} \;+\; \sum_{f\in\{C,\,E\}}\gamma_{6isf}\cdot F_{tf} + \;\gamma_{7is}\cdot N_{ist} \;+\; \sum_{q=1}^{4}\gamma_{8isq}\cdot Q_t \\ & +\sum_{d\in\{D,\,C\}}\gamma_{9isd}\cdot D_{istd} \;+\; \gamma_{10is}\cdot H_t \;+\; \sum_{l\in\{M,\,W\}}\gamma_{11isl}\cdot L_{istl} \;+\; \varepsilon_{ist}\end{aligned}$$

where:

$S_{ist}$ = sales of $i$ in store format $s$ and day $t$
$PI_{ist}$ = Price Index (price/average regular price) of $i$ in store format $s$ and day $t$
$B_{ist}$ = Baseline sales estimation of $i$ in store format $s$ and day $t$
$SI_{ist}$ = Seasonal Index estimation of $i$ in store format $s$ for day $t$
$M_{istm}$ = Dummy variable that takes the value 1 when $i$ is communicated through $m$ in store format $s$ and day $t$
$I_{ist}$ = Dummy variable assuming 1 when $i$ is in its initial promotional day
$F_{th}$ = Dummy variable that takes the value 1 when $t$ is in festivity period $f$[2]
$N_{ist}$ = Weighed percentage of stores selling $i$ in store format $s$ and day $t$
$Q_t$ = Dummy variable that assumes the value 1 when $t$ falls in month quarter $q$
$D_{istd}$ = Dummy variable that takes the value 1 when the promotion is type $d$[3]
$H_{ih}$ = Dummy variable that takes the value 1 when day $t$ is a weekday holiday, and -1 if it is adjacent to a weekday holiday
$L_{istl}$ = Dummy variable assuming 1 when $i$ last had a promotion $l$ time ago[4]
$\gamma_{is}$ = model coefficients
$e_{ist}$ = error component associated to $i$ in store format $s$, day $t$

[2] $C$ for Christmas, $E$ for Easter
[3] C for Card, D for Direct Discount
[4] W for week, M for Month

Estimating the model's coefficients through Elastic Net regressions involves minimizing the following function:

$$\min_{\gamma \in \mathbf{R^d}} \frac{1}{n} \|\mathbf{S} - \mathbf{X}\gamma\|_2^2 + \lambda_1 \sum_{j=1}^{d} |\gamma_j| + \lambda_2 \sum_{j=1}^{d} |\gamma_j|^2 \tag{4.1}$$

Where $\mathbf{S}$ is the sales vector, $\mathbf{X}$ is the input vector, $\gamma$ is the coefficient vector, $n$ the number of observations, $d$ the number of dimensions, and $(\lambda_1, \lambda_2)$ the weights attributed to the regularization terms. The overall importance of regularization is given by $\lambda = \lambda_1 + \lambda_2$. The parameter that governs the relative importance of each regularization term is defined as $\alpha = \frac{\lambda_1}{\lambda_1 + \lambda_2}$, $\alpha \in [0, 1]$. When $\alpha = 1$, elastic net becomes ridge regression, whereas $\alpha = 0$ it becomes LASSO. Notice that Equation 4.1 follows the form of the general formulation in Equation 2.1.

In their paper, Friedman et al. (2010) propose a methodology that involves cycling through decreasing values of $\lambda$ and selecting the one that best performs in cross-validation. For the purpose of this thesis, $\alpha$ will be defined first, and then $\lambda$ will cycle through $\lambda_0$, the value for the parameter that would render all coefficients zero, and a minimum $\lambda_{min}$ which is either zero or the first value of $\lambda$ that worsens the objective function.

**Random Forest** As described in Section 2.2.5, the Random Forests algorithm generates regression trees that grow on a random subset of the dataset and average their result. The variables used are the same as the linear model. In contrast with the linear model, however, where the natural log of sales and price indexes is taken, the variables are used as is. Applying a differentiable monotonic function to a continuous variable has no effect on the output when it comes to tree-like regressors.

In this algorithm, the hyperparameters to be tweaked are the number of trees built, the minimum number of observations in each leaf, and the sample rate, i.e., the percentage of the training dataset used to build each tree.

**Gradient Boosting Machine** Using the same logic as the Random Forests algorithm, a Gradient Boosting Machine was developed using the methodology described in Section 2.2.6. The hyperparameters used are exactly the same.

**Linear Model/Gradient Boosting Hybrid** As described in Chapter 2, tree-like models are apt for capturing interaction effects between variables, but not excellent at modelling continuous interactions. Linear models, on the other hand, would increase tremendously in size should variable interactions be included. For instance, given n predictors, the number of terms in a linear model that includes every predictor, and every possible interaction is $\binom{n}{0} + \binom{n}{1} + \binom{n}{2} + \cdots + \binom{n}{n} = 2^n - 1$. Since this quantity grows exponentially, it rapidly becomes impractically large.

An approach to reaping the advantages of both these methods was to build a hybrid model, where a linear model uses the continuous variables (Baseline Sales, Price Index, Seasonal Index and Store Quota) to make an initial prediction, which is afterwards corrected by a Gradient Boosting Machine that uses as input the remaining categorical variables.

All these models are available within the h2o R package. This package allows for parallel computing, where the same algorithm might be running in more than one core of a computer. For example, several trees of a Random Forest can be grown at the same time because of this feature.

In addition, after having some results from each model, the best performing ones were duplicated and adapted to have a base-times-lift approach. This approach assumes that promotional sales depend linearly of the baseline:

$$S_{ist} = F_{lift} \times B_{ist}$$

This approach renders the response variable $F$ dimensionless. This has the advantage that the model can extrapolate from sales data regardless of the quantity considered. This allowed the new models to be trained in data at the store level, rather than at the aggregate store format level, which makes the dataset about 80 times larger.

## 4.4 Evaluation and Tuning

In order to ultimately be able to pick, for each category, a model that performs reliably and satisfyingly, it is needed that:

**a)** The estimates of the model's accuracy are reliable;
**b)** The choice of hyperparameters converge to optimal.

To comply with these requirements, a model tuning, evaluation and selection framework was developed.

In the previous method of Batista (2016), as described in Chapter 3, when performing stepwise least squares regression, the model was trained once, and its variables selected by stepwise regression to maximize the performance in the entire training dataset. Even though the accuracy of the model is optimized in the training dataset, it does not necessarily maximize the predictive power of the model for future observations. In Chapter 2, we have seen that Kohavi (1995) describes a more reliable method for selecting an accurate model: that of cross-validation, where the training dataset is divided into $k$ fractions. The model is run $k$ times, once for every fraction that is left out of the training set. What is being optimized is not the performance of the model in the entire dataset, but the average performance in the "holdout" fractions. This gives us a more conservative and reliable estimate of the model's performance in observations it was never shown, and is, therefore, a superior choice of metric for model selection.

There is a small caveat to this process. The model's performance is given by the weekly MAPE. However, the models will predict daily observations. When performing cross-validation,

it's necessary to reflect the practical forecasting case as much as possible. It would not be realistic to split the dataset in a way that a few days from a given week would fall in the validation set and the remaining week in the training set. Therefore, cross-validation is performed $w$ times, $w$ being the number of weeks in the training set, and for every run, one of the respective weeks is excluded from the model. This way, cross-validation provides a realistic estimate for the model's performance when forecasting new weeks.

At the beginning of each run, a $n \times h$ random matrix of $n$ combinations of $h$ hyperparameters is generated. For each SKU/Store Format pair, the algorithm is ran for every combination in the matrix. In the end, the model with the highest cross-validation score is chosen. Table 4.3 clarifies the hyperparameters used for each model:

Table 4.3: Hyperparameters for each model

| Model | Hyperparameters |
|---|---|
| Linear Model | lambda, alpha |
| Random Forests | minimum leaf size, sample rate, number of trees |
| Gradient Boosting Machine | minimum leaf size, sample rate, number of trees |
| LM/GBM Hybrid | lambda, alpha, minimum leaf size, sample rate, number of trees |

Finally, to confirm that the algorithm is indeed converging and producing models with satisfactory performance, the models obtained predict the observations on the test set.

A challenge identified when executing this methodology was defining bounds for the hyperparameters. Therefore, after an initial run, results were compared with the hyperparameters used for each SKU/Store Format pair. If there were indications that a certain range for a given hyperparameter consistently produced bad results, the range within which that hyperparameter could vary was adjusted for a second run. The framework described above is outlined in Algorithm 1, present in Appendix B.

## 4.5 Product Interaction

With the intention of further exploring opportunities to improve forecasting accuracy, the possibility of including promotional information from other products was studied. In this section, a methodology to correct the models obtained via the method of Section 4.4 is described.

Batista (2016) divides product cannibalization into two categories: **a)** Promotional-Regular cannibalization, where promotional activity in some products affect the sales of unpromoted items, and **b)** Promotional-Promotional cannibalization, where the sales volume of two or more items promoted simultaneously is different than what they would have been should the promotions happen independently. The author claims that the latter is considerably harder to identify. However, it is the only object of study in this thesis, since only promotional observations are included in the dataset. The original model contained the subcategory promotional quota as a variable, which was removed.

In order to consider product interaction between products, it was decided to have an approach similar to the one present in Ma et al. (2016). While the authors use the LASSO Granger methodology to identify causality between promotional pressure and sales time series, the complexity of this methodology and time constraints of the project forced the choice of a simpler approach.

It was assumed that the items that had the most predictable influence within a category over all others were the ones with the highest sales volume and were promoted all year round. Therefore, an ABC/XYZ analysis was conducted for each category, with the following definitions:

- Ordering the SKUs by decreasing unit sales volume, the A group represents the first 80% of sales, the C group the last 5%, and the B group the remaining 15%;
- The XYZ grouping was made according to the coefficient of variation (CV = standard deviation/mean sales):
  - group X, if CV < 0.5,
  - group Y, if 0.5 < CV < 1, and
  - group Z, if CV > 1.

The items present in the AX group - the stablest and highest selling products - were assumed to contain a significant amount of information that could improve forecasts for the whole category.

The models with only the SKU's own predictors can, without exception, all be divided in a random term (Quantity $S$) being equal to a systematic term (Prediction $P$) plus a random error and be reduced to

$$\ln(S_{ist}) = \ln(P_{ist}) + \varepsilon_{1\,ist}$$

For each SKU in the AX group, its price index $PI$ and communication means $m$, if any, are extracted and used to explain the error term. To model product interaction, only a linear model was produced, and it takes the form:

$$\varepsilon_{1\,ist} = \ln(S_{ist}) - \ln(P_{ist}) = \sum_{\substack{j=1 \\ j \in A \cup X,\ j \neq i}}^{k} \left( \alpha_{ijs} \cdot \ln(PI_{jst}) + \sum_{m=1}^{4} \gamma_{ijsm} \cdot M_{jstm} \right) + \varepsilon_{2\,ist} \tag{4.2}$$

Given the interest in identifying few significant interaction pairs, the regularization parameter $\alpha$ was fixed at 1, producing LASSO regression. This minimizes the number of coefficients in the model, as shown in Chapter 2. The predictions from the optimal models obtained in Section 4.4 were used to obtain prediction errors, and the Random Search methodology was applied again, where the only hyperparameter subject to variation was the relative weight of the regularization penalty, $\lambda$.

# Chapter 5

# Results

In this chapter, the results obtained from applying the algorithms and methodology defined in the previous Chapter are shown. The methodology described in Section 4.4 is complemented with an example case. To emphasize the business relevance of the developed methodologies, an estimate of the added value is made according to the rule proposed by Kahn (2003). Further insight on the properties of the best performing models is given in Section 5.2. To keep the length of this section under reasonable limits, some of the plots were moved to Appendix C. Lastly, a brief description of a dashboard built with the purpose of monitoring forecasting and replenishment indicators is made in Section 5.3.

## 5.1 Forecasting results

Following the procedure defined in Chapter 4 and, more specifically, Algorithm 1, each model was ran at least once. For the initial runs, the results were analysed with respect to the hyperparameters chosen. To clarify this method with an example, we shall look closely at the first run of the Gradient Boosting Machine in the Chicken category, one of the first models ever run. At first, it was thought that few observations per leaf, and therefore very deep trees, would result in very accurate models. In this case, minimum observations per leaf could vary between 1 and 5. The results obtained can be observed in Figure 5.1.

As it can be observed, the resulting MAPE for each SKU-Store Format combination is not independent from the hyperparameters assigned to the models. In particular, the initial belief that very fine trees are better was disproved. To improve results, the minimum observations per node was allowed to vary up to as high as the number of observations would allow. This led to an improvement of the test MAPE from 22,5% to 21,7%. This learning was applied to all the remaining algorithms, by starting with very wide bounds for hyperparameters to vary at first, and narrowing down only after there is evidence a more reduced range improves results.

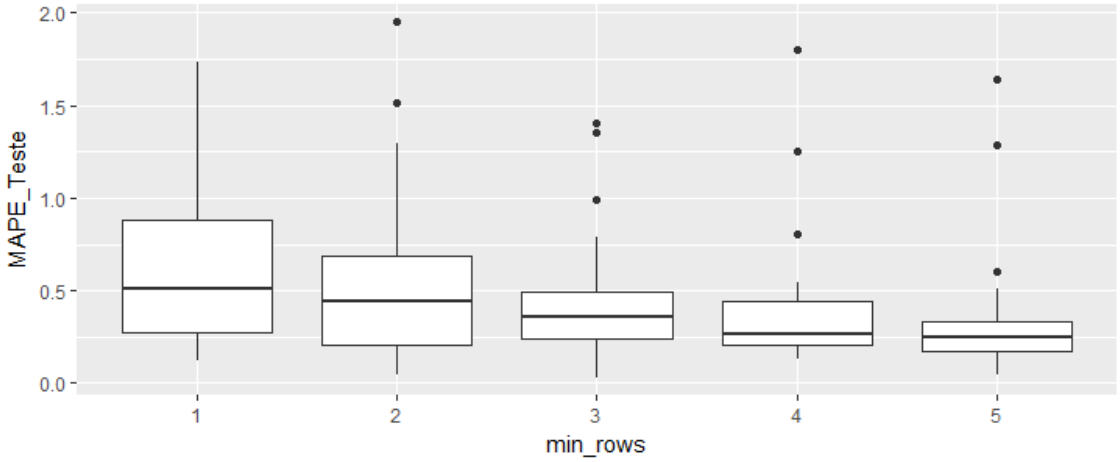



Figure 5.1: GBM initial results with respect to its hyperparameters

The exhaustive application of the developed procedures led to the results that figure in Table 5.1, where the best performing models are highlighted. The hardware used consisted of 16 cores of an Intel(R) Xeon(R) E5-2640, boasting 10 GB of RAM, 2.60 GHz of processing speed and 40 MB of cache.

The Base-times-Lift approach mentioned in Sections 2.2.4 and 4.3 was tried out for most category/algorithm combinations. However, the results were very unsatisfactory - most of the times worse than the results of the previous model - and therefore not included in this chapter. The explanation found for these results is that the Base-times-Lift approach would be more appropriate for a more aggregate analysis of sales data. For example, a store that sells on average one unit of a certain product every week will have a daily baseline of around $^1/_7$ units. However, there are no observations for days with zero sales, which means that the store's average lift is 7. Being able to extrapolate the sales of the whole company from the reactions of individual stores to promotions becomes very unlikely.

Table 5.1: Results

| | Cross Validation Metrics | | Test Period | | Average hyperparameter values | | | | | Runtime |
|---|---|---|---|---|---|---|---|---|---|---|
| | MAPE (%) | Bias (%) | MAPE (%) | Bias (%) | $\alpha$ | $\lambda$ | #trees | Sample ratio | Leaf size | (minutes) |
| *Chicken Category* | | | | | | | | | | |
| Linear Model | 18.26 | −2.73 | 25.21 | −4.33 | 0.53 | 1.3e−3 | | | | 12.2 |
| Random Forests | 17.48 | −0.02 | 23.13 | −5.40 | | | 56.1 | 0.76 | 1.77 | 21.3 |
| G. Boosting Machine | 17.21 | 0.46 | 21.75 | −7.56 | | | 26.2 | 0.58 | 3.44 | 12.3 |
| **LM/GBM Hybrid** | 18.38 | −5.32 | **20.70** | −9.34 | 0.50 | 7.9e−5 | 34.9 | 0.55 | 12.62 | 23.3 |
| *Fruits Category* | | | | | | | | | | |
| Linear Model | 20.40 | −2.14 | 40.21 | 0.46 | 0.50 | 1.8e−2 | | | | 50.4 |
| **Random Forests** | 18.95 | 0.84 | **24.91** | 3.89 | | | 55.6 | 0.77 | 2.42 | 39.3 |
| G. Boosting Machine | 18.50 | 0.88 | 27.00 | 3.01 | | | 47.9 | 0.80 | 7.53 | 42.4 |
| LM/GBM Hybrid | 21.91 | −4.55 | 35.10 | −8.46 | 0.30 | 8.8e−3 | 31.7 | 0.70 | 9.80 | 54.3 |
| *Frozen Fish Category* | | | | | | | | | | |
| Linear Model | 33.22 | −7.35 | 37.29 | −2.09 | 0.65 | 1.7e−4 | | | | 67.6 |
| Random Forests | 31.07 | −0.25 | 32.40 | 4.94 | | | 54.7 | 0.74 | 2.15 | 47.3 |
| G. Boosting Machine | 29.75 | −1.15 | 33.28 | 6.00 | | | 48.7 | 0.80 | 15.38 | 55.5 |
| **LM/GBM Hybrid** | 30.48 | −14.93 | **31.02** | −13.16 | 0.25 | 3.1e−4 | 33.2 | 0.71 | 18.16 | 92.9 |

After ascertaining which is the best algorithm for each category, the method described in Section 4.5 was applied to each of them. The result of the corrections via product interactions are listed in Table 5.2.

Table 5.2: Results after including product interaction

| | Best Model | MAPE before correction | MAPE after correction | Diff |
|---|---|---|---|---|
| Chicken | LM/GBM Hybrid | 20.70% | 20.55% | -0.15pp |
| Fruit | Random Forests | 24.91% | 24.60% | -0.31pp |
| Frozen Fish | LM/GBM Hybrid | 31.02% | No improvement | - |

The improvements in accuracy obtained by including product interaction are unfortunately quite dismal. Nevertheless, another series of plotting functions were made to visualize how product sales are affected by the promotional properties of others, which will be shown in the next section.

To conclude this section, we shall compare the final results of the methodologies developed with the forecasting accuracies of the previous model. Furthermore, the estimation rule proposed by Kahn (2003) will be applied to each Category in order to have a grasp of the potential savings should this methodologies be implemented. Assuming that the retailer has a profit and gross margin equivalent to the whole retail industry, as seen in Chapter 1, 3% and 17% respectively, that the cost of goods sold is therefore (100% - 17%) = 83% of net sales, and that the occurrence of underestimation is just as likely as overestimation, the reduction in cost of forecasting error can be given by

$$\frac{\Delta_{error} \times (83\% + 3\%) \times Net\ Sales}{2}$$

A summary of this analysis is found in Table 5.3.

Table 5.3: Comparison of results and estimate of business impact

| | Previous MAPE | New MAPE | Reduction | Savings as a % of Yearly Promotional Net Sales Volume |
|---|---|---|---|---|
| Chicken | 38.8% | 20.6% | 18.2pp | $\sim$ 7.8% |
| Fruit | 30.5% | 24.6% | 5.9pp | $\sim$ 2.5% |
| Frozen Fish | 37.1% | 31.0% | 6.1pp | $\sim$ 2.6% |

Of course, the savings estimate presents itself as a ceiling value for the actual savings. Kahn (2003) assumes that all units underforecast convert into lost sales, which is not true since each product has its substitutes. Moreover, not every unit overforecast is spoiled, and can be sold at a markdown some days later.

## 5.2 Properties of the models obtained

A series of plotting functions were constructed to facilitate the evaluation of each algorithm's results, the importance given to each variable and the strength of interaction between products. Firstly, for each individual product, it is possible to analyse the forecast performance of the model in the training and test period.

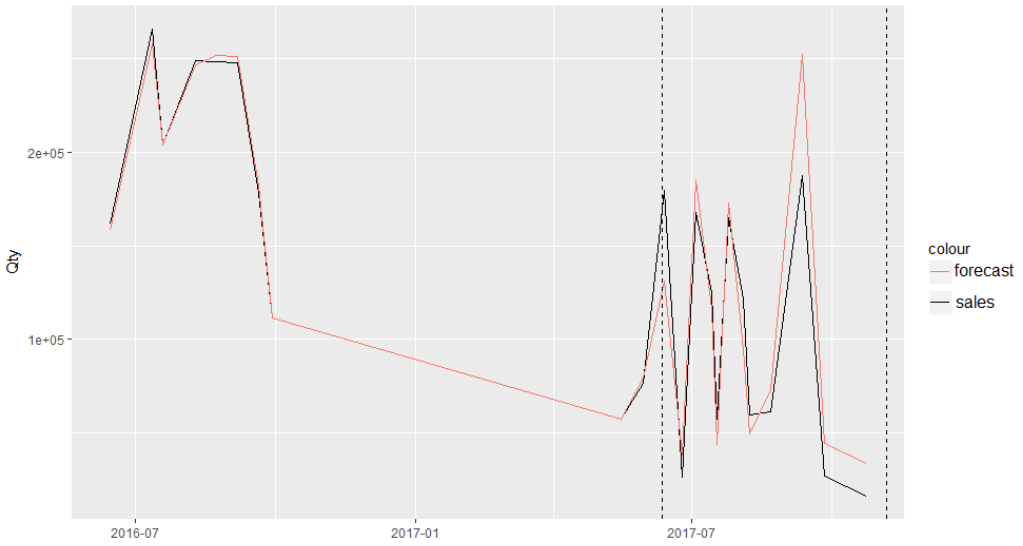



Figure 5.2: Sales and Forecast comparison for a given product

Notice how in Figure 5.2 the model was able to accurately predict sales with very few observations in the training period. A scatter plot, as is the example of Figure 5.3, is a more appropriate visualization to analyse the dispersion of error.

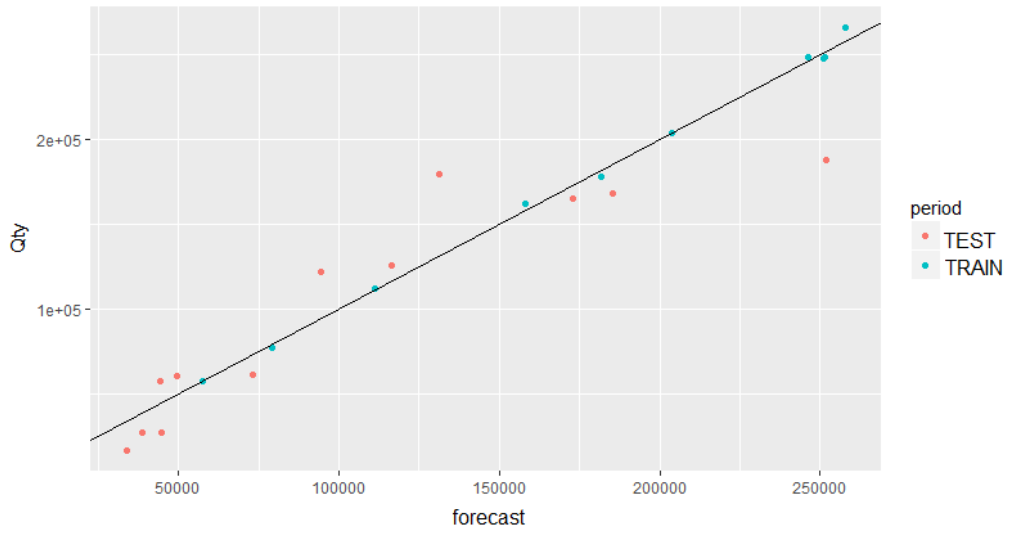



Figure 5.3: Sales and Forecast scatter plot

It is also possible to analyse the response of sales respective to independent variables. This allows to ascertain which variables have a significant impact on sales for each individual product and if the input variables are similar in the training and testing period. Scatter plots were used for continuous variables, whereas box-plots are more appropriate for categorical ones.


Figure 5.4: Sales response of a given product to its continuous input variables

For this particular product, baseline sales are clearly the most important variable, and it is unclear whether it reacts significantly to deeper price reductions.

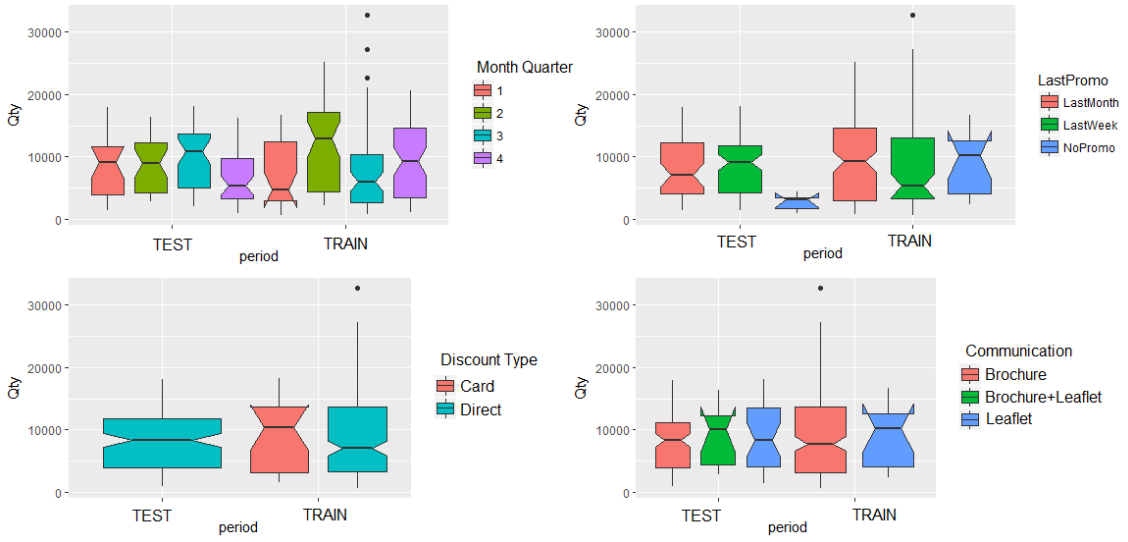



Figure 5.5: Sales response of a given product to some of its categorical input variables

These plots might shed some light over, for instance, which is the most appropriate communication means for the specific product, and whether too many promotions provoke a saturation effect. In Figure 5.5, the notch in each box-plot represents a confidence interval for the median sales. If two notches do not overlap there is 95% confidence their medians differ.

The visualizing of the overall progression of MAPE throughout the training and test periods can also provide useful insight, such as whether some specific part of the year is harder to predict. In Figure 5.6, the test period is marked with vertical dashed lines. In the train period, cross validation predictions are used. In this chart, it would seem that Winter sales have a relatively lower prediction error when compared to late Summer and Autumn.

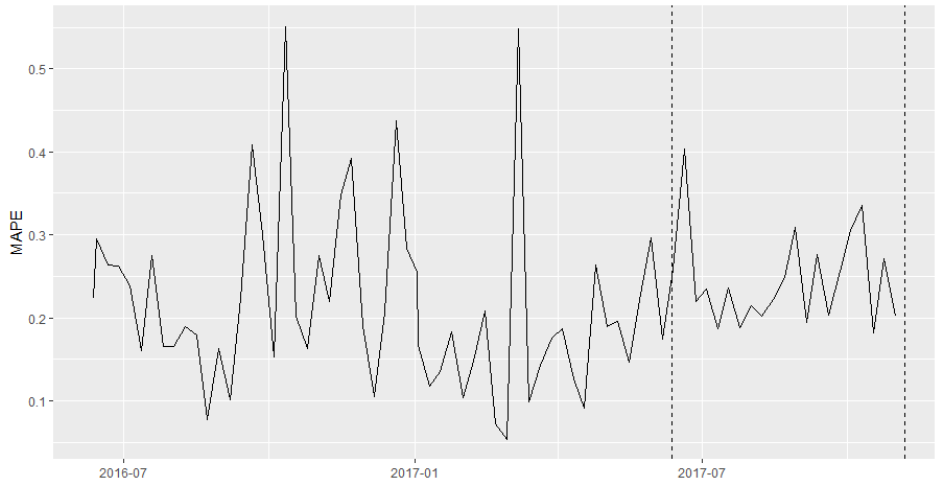



Figure 5.6: Evolution of MAPE for the Fruits category

An important insight also provided by the developed plotting functions is the relative importance of variables. A linear model is very transparent in this regard, since it is possible to also know the size of its coefficients. In tree-like models there is not a metric for the impact of a certain variable on the end result. However, variable importance can still be computed by comparing the number of times a certain variable was used to split the data by the algorithm. For each category, a chart with variable importance (and coefficients, if the model is linear or hybrid) is produced. Figure 5.7 shows the distribution of variable importance for the Fruits Category. As expected, the baseline is considered to be most important by most models, followed by price and seasonal index.

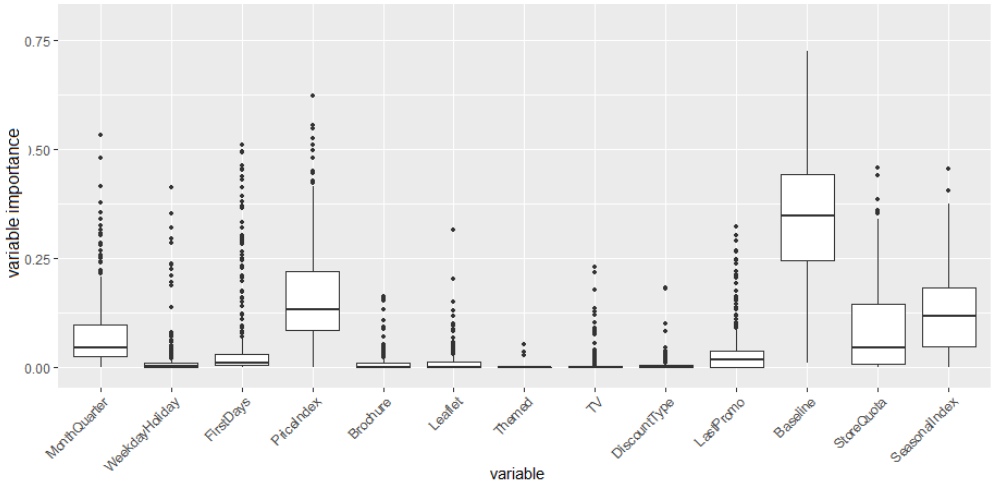



Figure 5.7: Relative variable importance in the Fruits Category

Even though the percentage decrease in MAPE after correcting the models with product interaction is not extraordinary, it is still an improvement. We have seen that a small improvement in forecasting error can have a notorious reduction in cost if the sales volume is high. Moreover, product cross-elasticities are useful for business tasks other than forecasting. For example, it is useful to know which items are complementary or substitutes when deciding a store's assortment, how much shelf space to allocate to each product, which products (not) to place next to each other, and which items not to promote simultaneously.

To quickly identify the interactions between products, the coefficients obtained by LASSO regression are plotted in a single axis. If we have a look at Equation 4.2, we can see that a positive coefficient for the price index of a given product means that a discount in that product results in a reduction of the sales prediction for the product being modelled, meaning they are substitutes. However, a positive coefficient for a given communication means $m$ would correct the prediction upwards. Therefore, we take the negative of the coefficients for communication means and plot them, together with the price index coefficients, in a single axis. An example of such a plot is visible in Figure 5.8.

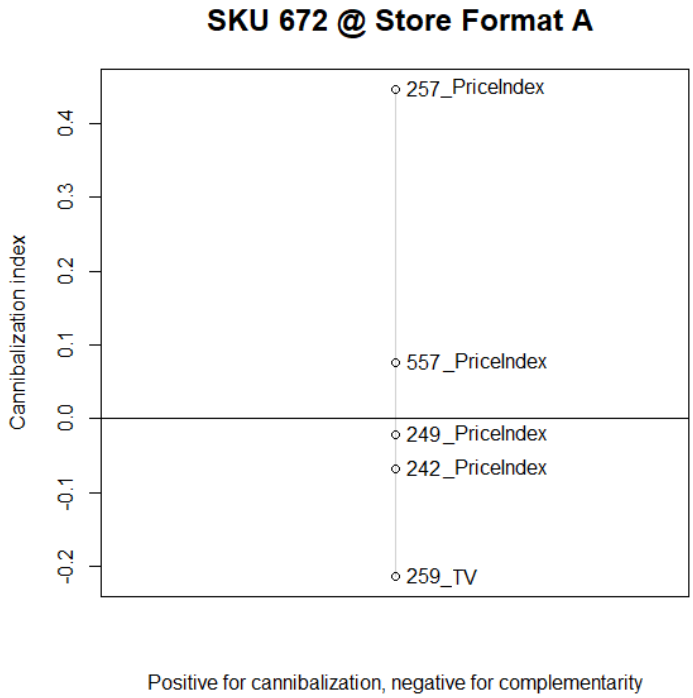



Figure 5.8: Example of a product interaction plot

Figure 5.8 details which products from the AX group have the most effect on sales of the product in question. In this particular case, for store format A, a discount in product 257 is expected to have a negative impact on the sales of 672, whereas historical TV campaigns of product 259 are estimated to have been responsible for an increase in sales of the same product.

## 5.3 Visualization

With the purpose of monitoring the project's indicators and showing results to the forecasting and stock management team, a dashboard containing several perspectives was developed in MicroStrategy Desktop.

The dashboard consists of indicator panels where sales, forecasting indicators (such as MAPE and Bias) and replenishment indicators (Expected Lost Sales and Stock Coverage) can be analysed at different levels of aggregation, both of the product and store hierarchy as well as at different time frames. This way, a stock manager, a supervisor, and a member of the project team can all benefit from the dashboard and draw insights from the data. The dashboard consists of 7 panels, as follows:

1. Weekly evolution of sales, MAPE and Bias in detail in Figure 5.9;
2. Weekly evolution of missed sales and stock coverage;
3. Analysis of the average week;
4. Stock coverage control at the SKU level;
5. Missed sales control at the SKU level;
6. Daily SKU indicators;
7. Service Level/Stock Coverage trade-off analysis.

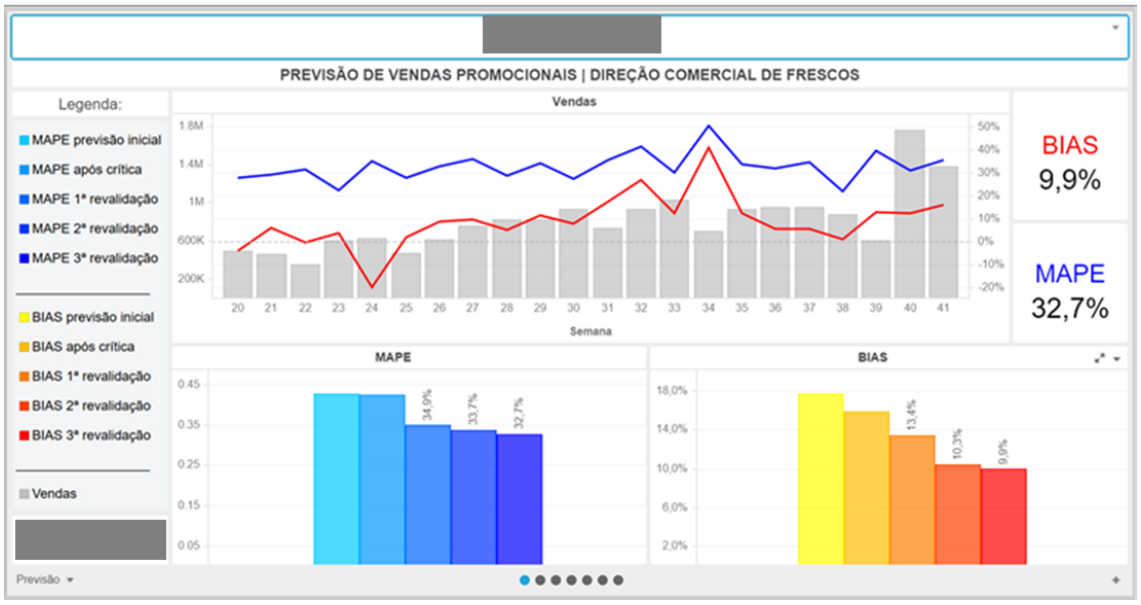



Figure 5.9: Forecasting indicator panel of the dashboard

A screenshot and a more detailed description of every panel can be found in Appendix D.

# Chapter 6

# Conclusion

This closing chapter begins by enumerating the main takeaways from the work developed. The second section consists of a reflection on the importance of forecasting culture and how good forecasting algorithms alone are not enough. Lastly, a self-critic on this work is made, as well as a suggestion of possible extensions of this thesis.

## 6.1 Implications for practice

The key takeaway from this work is that advanced algorithms and machine learning techniques have a superior performance when compared to traditional regression methods. We have also seen that there is not a master algorithm that will perform the best in all cases. The algorithms tested responded differently to the sales patterns of each category. Whereas linear models behave satisfactorily when sales are relatively stable, as is the case of the Chicken category, when sales have very seasonal patterns - where some input variables fall outside the spectrum the model was trained on - a more robust model performs better.

Experimenting several models is interesting from a knowledge discovery and academic point of view. However, it would be sensible that a business application relied on only one, for consistency and simplicity. Despite not being the best performing algorithm in all categories, the Random Forests algorithm proved to be the best all-rounder. Given its nature, a Random Forest will never place predictions outside the range of values it was trained on, in contrast with all the other methods, which gave it the edge in the quite irregular Fruits category.

A disadvantage of tree models when compared with linear models is the fact that the former are less interpretable. While it is still possible to compare variables as to their relative importance, we no longer know the elasticity of the response variable regarding to the inputs. Therefore, estimating which direction sales will go when adjusting a certain variable is not an option.

Moreover, there is a caveat when switching from a linear model to a more complex regression algorithm. While the export format of a linear model is very simple, consisting merely of the coefficients that compose it, a tree-like model, especially if it is an ensemble such as one like a Random Forest or a Gradient Boosting Machine, can be composed by thousands of conditions,

which make it very hard to be exported and used in a common spreadsheet software, like Microsoft Excel. Therefore, deploying these kinds of models for business use would require either the development of a web based application, or computing these calculations outside of the spreadsheet using a middleware application.

Many algorithms come with default values for their hyperparameters. We have seen that the performance of the algorithms is sensitive to the choice of hyperparameters. Therefore, conducting a random search for these parameters is likely to improve results. This thesis serves as both a practical application of the work of Bergstra and Bengio (2012), as well as an extension of it, since the methodology used combines random searching through hyperparameters with sequentially narrowing down their range.

Unfortunately, the improvements obtained from correcting the models with promotional information of other products were below expectations. It would seem that the variation that could eventually be explained by product interaction is shadowed by either noise or variation from other sources. Suggestions of other variables to explore are given in Section 6.3.

Lastly, there are learnings from this thesis that can be taken to processes other than sales forecasting. For example, a more reliable estimate of forecasting error will allow for better calculations of safety stocks, and for a more educated guess on which service levels to assign for each product. The computed elasticities for the several variables can be used to better decide which products to promote through each of the communication means, or with which price cut.

## 6.2 A reflection on forecasting culture

Imagine a situation where it would be possible that we know exactly what and when our customers are going to buy. However, if the supply chain is not able to act accordingly to this information, there still is going to be lost sales, and overstocking at some echelon of the chain. Therefore, even if we have the most powerful forecasting model available with the current technology, in order to reap the benefits of such a model, the supply chain itself must also be able to carry on the forecasting accuracy through its several stages. Moreover, the quality of a model will only be as high as the data given to it as input.

I naturally had the opportunity of spending quite some time with stock managers and planners of the Fresh Division, as well as their supervisors. Even though the goal of this dissertation was and is to produce a better forecasting model for promotional sales and integrate cannibalization in such model, initiatives directed at developing the retailer's forecasting culture would perhaps have an even greater impact on performance.

The underlying mathematics behind a forecasting model is merely one element among many that contribute to good forecasting accuracy. During my dissertation period I worked on other parallel initiatives, among them the development of an evaluation matrix for both the performance of the planners themselves, but also the quality of the processes in each business unit as a whole.

We identified several elements other than the underlying technology that were relevant for forecasting and replenishment performance. In the light of this paper, these criteria can be aggregated into two groups. The first can be described as the conditions that the model itself needs in order to perform properly. These criteria would be the quality of the data used as input, both in terms of recency and faithfulness to reality.

The latter, broader group can be described as the conditions in which the teams perform the manual tasks that the model can not be expected to perform, such as validating predictions, accounting for variables that the model does not consider, and aligning replenishment parameters to the company's policy. Examples of these criteria are the level of automation of the tools used, the level of knowledge of the planner, the quality of the communication between the several stakeholders, alignment of objectives and the visibility over historical data.

The constructed dashboard aims to provide more visibility over business indicators and hopefully increase the presence of evidence-based decision making at the retailer.

## 6.3 Future research

Despite the significant improvements in forecasting performance, the contents of this thesis are not without its limitations, nor are improvement opportunities exhausted.

When it comes to the data used, ample literature provides support for other business variables to be considered when forecasting. Zanders (2016) quantified the impact of weather in sales, and his work is particularly relevant when it comes to "deweatherizing" sales data. Many other authors cited in this thesis, especially those mentioned in Section 2.3.4, include store display conditions in their models, i.e. whether the product is featured in any special way. The fact that, in the retailer being studied, this is a decision that is left to the stores to make and is not recorded, made it impossible for this aspect to be included in the analysis. Lastly, it is a fact that many promotions are made entirely as a reaction to other retailers' promotions. Systematizing a process to collect promotional activity from competitors, transforming that information into variables and then including such variables in the models would certainly add value to the forecasting process.

A straightforward way to improve forecasting performance would be to correct the predictions in the test period with the identified Bias when cross-validating. As seen in Table 5.1, the test Bias does not always have the same signal as the cross-validation Bias. However, these are aggregate metrics. Correcting each SKU case-by-case would likely improve the end result nonetheless.

As already mentioned in Section 4.5, the methodology to incorporate product interaction developed by Ma et al. (2016) was not implemented in full. In order to have cannibalization models that can reduce MAPE more effectively, the proposed LASSO-Granger methodology should be applied to the products in each category, and perhaps across categories, to identify SKU pairs that show evidence of interacting, instead of the brute-force approach used in this thesis. Alternatively, the approach of Hruschka et al. (1999) might also provide some insight into product interaction. Deducting complementarity and cannibalization from whether products show up together in the

same sales basket more or less often, respectively, might be the key to identifying significant interaction pairs. Product substitution is relevant not only for forecasting, but even more so for assortment and space allocation decisions.

Despite the relatively long section on concept drift (Section 2.2.7.3), the only measure taken to tackle it was to eliminate older data from the analysis. To more successfully approach hypothetical concept drift, further analysis on sales trends would be necessary.

The application of a common practice, the Base-times-Lift approach, failed almost completely. The justification found is the fact that, in this thesis, the opposite was done of what is often seen in literature: computing lift factors with more aggregate data. This thesis' approach consisted of predicting SKU/Store Format sales from even less disaggregated data, which proved to be unfruitful. It is recommended to experiment the Base-times-Lift approach aggregating data using both the store and product structure.

Another characteristic of the models that proved to be an obstacle was the incoherence between the level of aggregation of the observations (daily sales) and the accuracy metric (weekly MAPE). It was found that optimizing the models to predict daily sales does not necessarily improve weekly accuracy. A more coherent approach would be to transform every promotional campaign into a weekly campaign using the typical sales volume of each day of the week. This would allow observations to be aggregated at the weekly level.

Lastly, looking back at how promotional sales were forecast even before the work of Batista (2016), there might have been other algorithms that could have been able to be successfully integrated in this context. The initial methodology was based on finding the most similar campaign to the one to be predicted, for each SKU. An approach could have been to systematize this methodology with proper machine learning "etiquette". For example, the k-Nearest Neighbours algorithm, or kNN, introduced by Altman (1992), maps observations in the feature space. For every new prediction made, the algorithm identifies the k nearest previous observations according to their "distance" in the feature space, and averages the sales quantity of those k neighbours, or alternatively, performs a weighed average based on their distance. This approach would have been able to resemble the initial method a lot more, which would reduce the disruptiveness of the subsequent implementation.

# Bibliography


Aburto, L. and Weber, R. (2007). Improved supply chain management based on hybrid demand forecasts. *Applied Soft Computing*, 7(1):136–144.

Ailawadi, K. L., Harlam, B. A., César, J., and Trounce, D. (2006). Promotion profitability for a retailer: The role of promotion, brand, category, and store characteristics. *Journal of Marketing Research*, 43(4):518–535.

Akaike, H. (1974). A new look at the statistical model identification. *IEEE Transactions on Automatic Control*, 19(6):716–723.

Ali, O. G., Sayin, S., Woensel, T. v., and Fransoo, J. (2009). Sku demand forecasting in the presence of promotions. *Expert Systems with Applications*, 36(10):12340–12348.

Altman, N. S. (1992). An introduction to kernel and nearest-neighbour nonparametric regression. *The American Statistician*, 46(3):175–185.

Arunraj, N. S. and Ahrens, D. (2015). A hybrid seasonal autoregressive integrated moving average and quantile regression for daily food sales forecasting. *International Journal of Production Economics*, 170:321–335.

Batista, B. (2016). Análise quantitativa de encomendas e efeitos promocionais em produtos perecíveis. *Faculdade de Engenharia da Universidade do Porto*, Master's Thesis.

BDO (2016). Comércio a retalho, exceto de veículos automóveis e motociclos - análise setorial - novembro 2016. Report.

Bergstra, J. and Bengio, Y. (2012). Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13:281–305.

Bishop, C. M. (2006). *Pattern Recognition and Machine Learning*. Information Science and Statistics. Springer.

Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons, 5th edition.

Breiman, L. (1996). Bagging predictors. *Machine Learning*, 24(2):123–140.

Breiman, L. (2001). Random forests. *Machine Learning*, (45):5–32.

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and regression tree*. Wadsworth International Group, Belmont, CA.

Cooper, L. G., Baron, P., Levy, W., Swisher, M., and Gogos, P. (1999). Promocast™: A new forecasting method for promotion planning. *Marketing Science*, 18(3):301–316.

Dawes, J. G. (2012). Brand-pack size cannibalization arising from temporary price promotions. *Journal of Retailing*, 88(3):343–355.

Dekker, M., van Donselaar, K., and Ouwehand, P. (2004). How to use aggregation and combined forecasting to improve seasonal demand forecasts. *International Journal of Production Economics*, 90(2):151–167.

Derks, L. (2015). Improving promotion forecasts for the dutch fmcg market. *Eindhoven University of Technology*.

Domingos, P. (1999). *The Role of Occam's Razor in Knowledge Discovery*, volume 3, pages 409–425.

Donselaar, K. H. v., Peters, J., de Jong, A., and Broekmeulen, R. A. C. M. (2016). Analysis and forecasting of demand during promotions for perishable items. *International Journal of Production Economics*, 172:65–75.

Friedman, J., Hastie, T., and Tibshirani, R. (2010). Regularization paths for generalized linear models via coordinate descent. *Journal of statistical software*, 33(1):1.

Friedman, J. H. (2002). Stochastic gradient boosting. *Computational Statistics & Data Analysis*, 38(4):367–378.

Gama, J., Žliobaitė, I., Bifet, A., Pechenizkiy, M., and Bouchachia, A. (2014). A survey on concept drift adaptation. *ACM Computing Surveys (CSUR)*, 46(4):44.

Geman, S., Bienenstock, E., and Doursat, R. (1992). Neural networks and the bias/variance dilemma. *Neural Computation*, 4:1–58.

Gonçalves, J. (2000). *Gestão De Aprovisionamentos: STOCKS - PREVISÃO - COMPRAS*. Publindustria.

Heizer, J. and Render, B. (2006). *Operations Management*. Pearson Prentice Hall, 8th edition.

Hoch, S. J., Kim, B.-D., Montgomery, A. L., and Rossi, P. E. (1995). Determinants of store-level price elasticity. *Journal of Marketing Research*, 32(1):17–29.

Hruschka, H., Lukanowicz, M., and Buchta, C. (1999). Cross-category sales promotion effects. *Journal of Retailing and Consumer Services*, 6:99—105.

Huang, T., Fildes, R., and Soopramanien, D. (2014). The value of competitive information in forecasting fmcg retail product sales and the variable selection problem. *European Journal of Operational Research*, 237(2):738–748.

Instituto Nacional de Estatística, I. P. (2017). Estatísticas do comércio 2016.

Kahn, K. B. (2003). How to measure the impact of a forecast error on an enterprise? *The Journal of Business Forecasting Methods & Systems*, 22(1):21–25.

Kohavi, R. (1995). A study of cross-validation and bootstrap for accuracy estimation and model selection. In *International Joint Conference on Artificial Intelligence*.

Koottatep, P. and Li, J. (2006). Promotional forecasting in the grocery retail business. *Massachusetts Institute of Technology*.

Kourentzes, N. and Petropoulos, F. (2016). Forecasting with multivariate temporal aggregation: The case of promotional modelling. *International Journal of Production Economics*, 181:145–153.

Kuo, R. (2001). A sales forecasting system based on fuzzy neural network with initial weights generated by genetic algorithm. *European Journal of Operational Research*, 129(3):496 – 517.

Laan, M. v. d., Polley, E. C., and Hubbard, A. E. (2007). Super learner.

Liu, Y., Ren, P., Zhao, T., Yang, Z., and Gao, J. (2016). Estimation of adjacent substitution rate based on clustering algorithm and its application. In *12th International Conference on Natural Computation, Fuzzy Systems and Knowledge Discovery*, pages 794–800.

Ma, S., Fildes, R., and Huang, T. (2016). Demand forecasting with high dimensional data: The case of sku retail sales forecasting with intra- and inter-category promotional information. *European Journal of Operational Research*, 249(1):245–257.

Miranda, T. (2017). Sonae queixa-se de excesso de "promoções" por parte da concorrência. *Expresso*.

Narasimhan, C., Neslin, S. A., and Sen, S. K. (1996). Promotional elasticities and category characteristics. *Journal of Marketing*, 60(2):17–30.

Ramanathan, U. and Muyldermans, L. (2011). Identifying the underlying structure of demand during promotions: A structural equation modelling approach. *Expert Systems with Applications*, 38(5):5544–5552.

Tibshirani, R. (1996). Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B*, 58(1):267–288.

Vindevogel, B., Van de Poel, D., and Wets, G. (2004). Investigating the cross-sales effect of product associations.

Wittink, D. R., Addona, M. J., Hawkes, W. J., and Porter, J. C. (1988). Scan* pro: The estimation, validation and use of promotional effects based on scanner data. *Internal Paper, Cornell University*.

Zanders, S. C. W. (2016). The quantified impact of weather on fmcg sales. *Eindhoven University of Technology*, Master's Thesis.

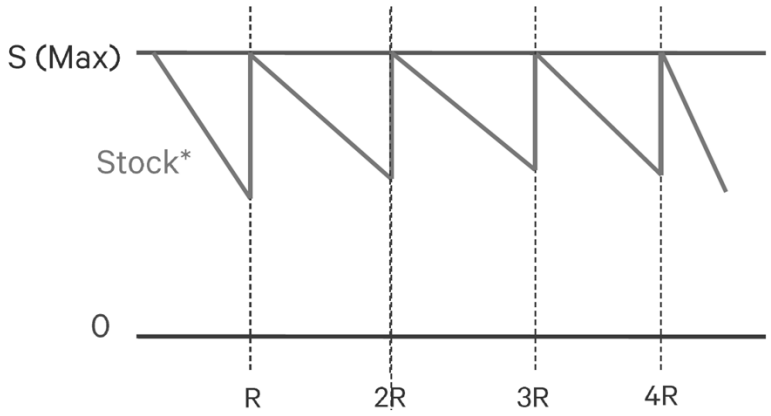
Zou, H. and Hastie, T. (2005). Regularization and variable selection via the elastic net. *J. R. Statist. Soc. B*, 67(2):301–320.

# Appendix A

# The constant replenishment model

The constant replenishment method, also known as the (R, S) model, is one of the simplest ones and consists of orders being placed at every stock revision cycle (R), to meet a desired inventory position (S).

This model is explained visually in Figure A.1.




*Includes In-Transit Stock

Figure A.1: The constant replenishment model

# Appendix B

# Model tuning and selection procedure

```
for Category do
    input     : Respective sales dataset
    Split dataset in train/test;
    for Algorithm a do
        while Hyperparameter range not OK do
            define    : Hyperparameter ranges
            generate: n Hyperparameter combinations
            for Base Unit; SKU do
                for Store Format do
                    for Hyperparameter combination h do
                        for i ∈ weeks do
                            Run algorithm a with parameters h holding week i out;
                            Compute score in week i;
                        end
                        Average all scores;
                    end
                    Pick combination with best average score;
                    Train best model in entire training set;
                    Predict observations in test set;
                end
            end
            Compute summary metrics for entire category;
            Analyse hyperparameter influence;
            if Hyperparameter range OK then
                end while
            end
        end
    end
    Compare cross-validation and test set results between algorithms;
    Select best algorithm;
end
```

**Algorithm 1:** Model tuning, evaluation and selection framework

# Appendix C

# Algorithm graphical outputs

To keep the body of the thesis at a reasonable size, the evolution of MAPE and variable importance of the Chicken and Frozen Fish categories figure in this appendix.

In Figure C.1 it is visible the general "unseasonality" of this category. Predictability and, assumingly, sales patterns, are relatively constant throughout the year.

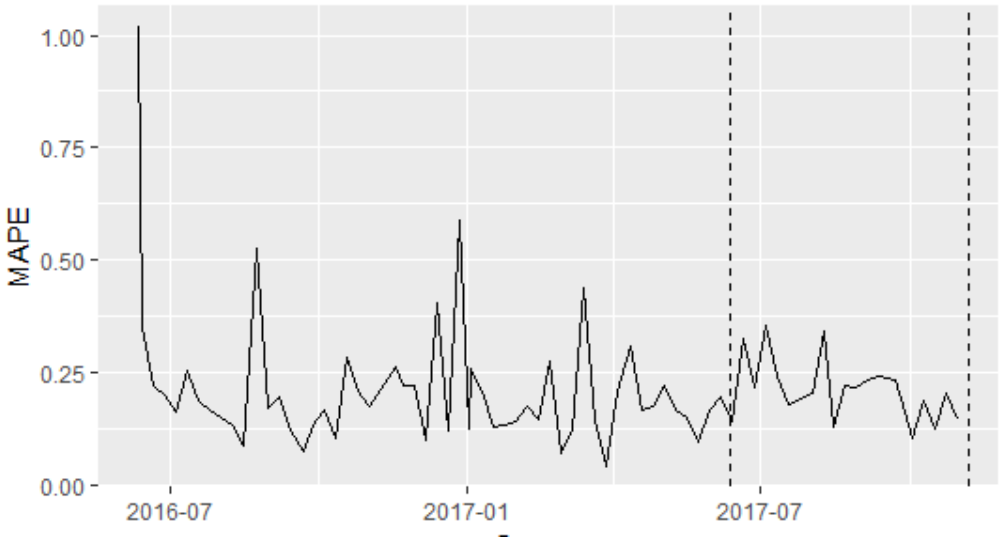



Figure C.1: Evolution of MAPE for the Chicken category

Figure C.2 describes the expected scenario where an increase in baseline sales, store quota and seasonal index increase sales, whereas increasing the price reduces them.

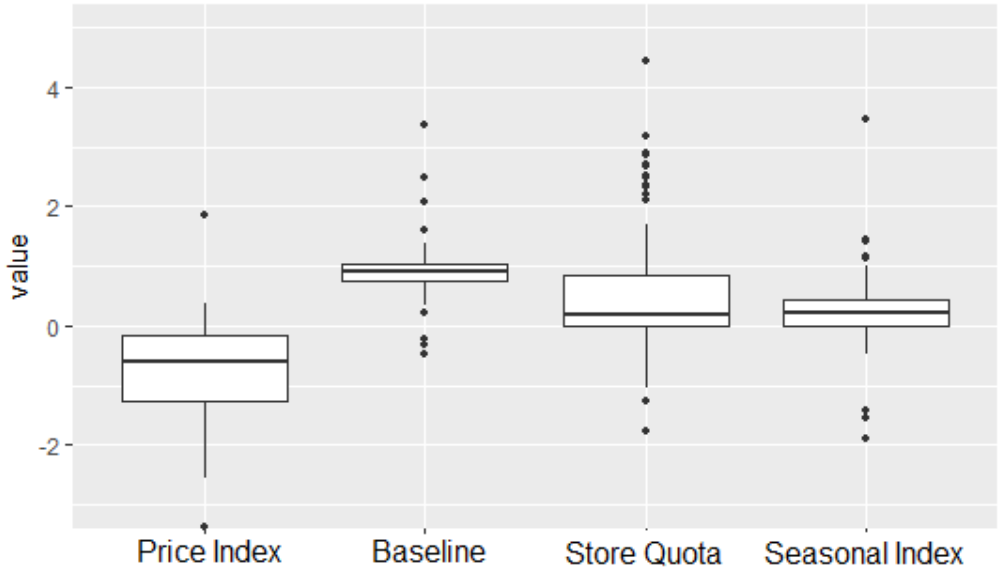



Figure C.2: Coefficients of the continuous variables in the Chicken category

As to the other, categorical variables, the ones most commonly used by the GBM model to explain sales is whether the promotion is in its initial days, followed by the TV and Brochure communication vehicles. It would seem customers are responsive to promotions in this category.

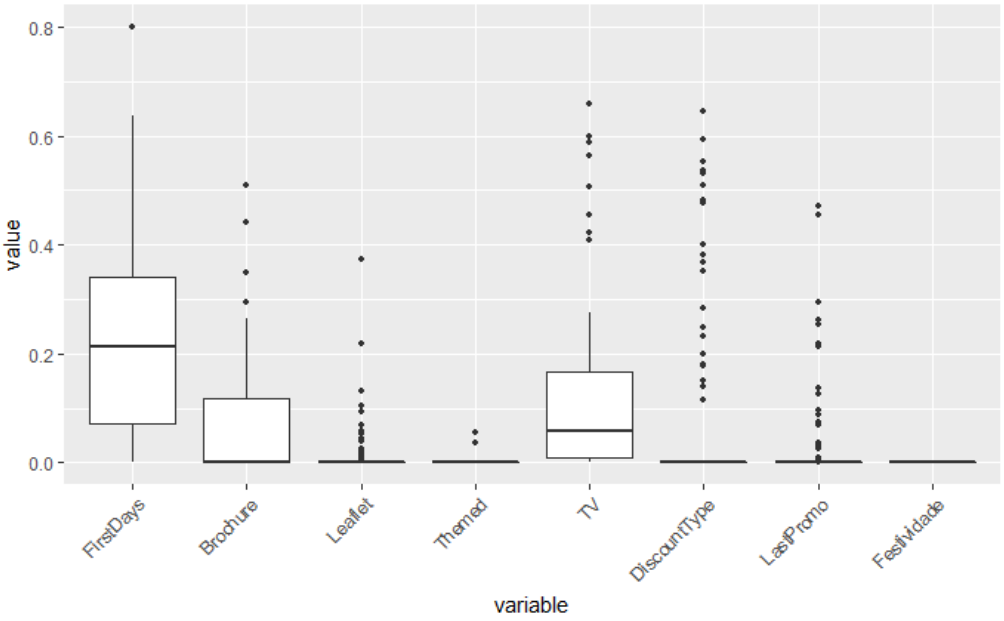



Figure C.3: Variable importance of the categorical variables in the Chicken category

Frozen Fish is the category with the highest MAPE. Figure C.4 would suggest that sales during Christmas and Summer are harder to predict. Forecast error varies significantly throughout the whole year.

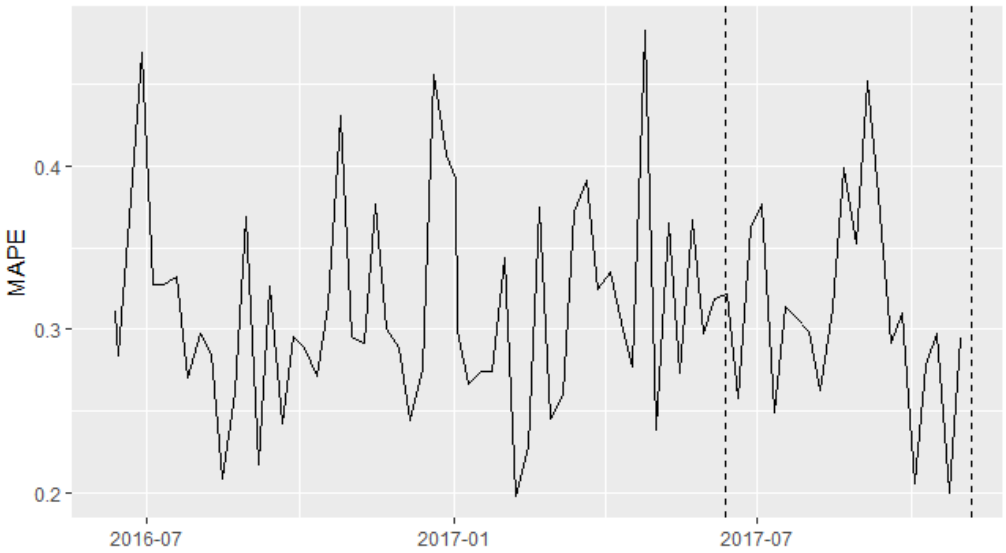



Figure C.4: Evolution of MAPE for the Frozen Fish category

As to the continuous variables used to explain sales in the Frozen Fish category, the situation is similar to the other categories. By comparing Figures C.5 and C.2, it would seem customers are more reactive to the price discount when buying fish.

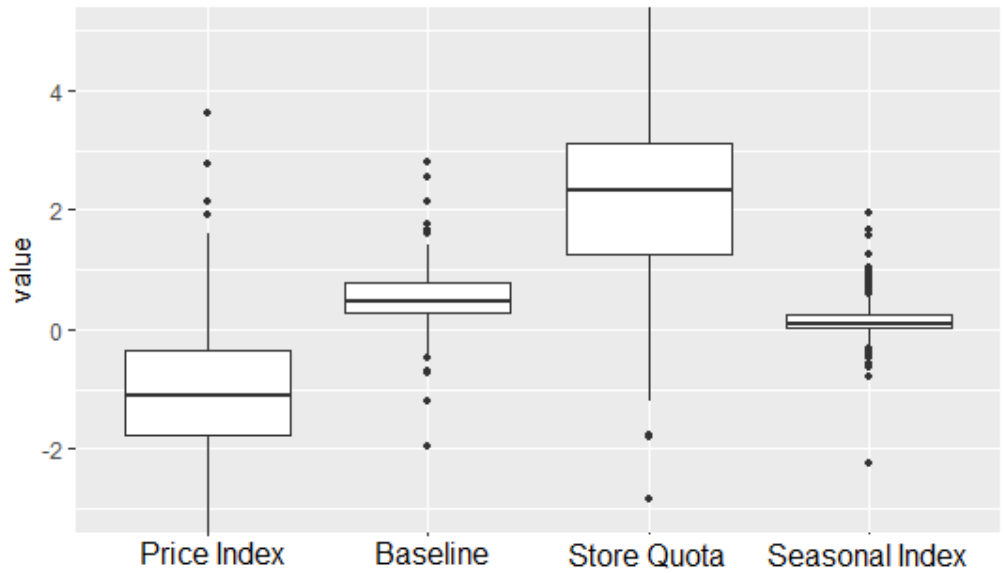



Figure C.5: Coefficients of the continuous variables in the Frozen Fish category

The fact that frozen fish has a very high shelf life compared to other categories enables customers to buy in large quantities when the prices are low, regardless of the communication means. This thesis is supported by Figure C.6, as the time since last promotion is the most relevant categorical variable to model sales.

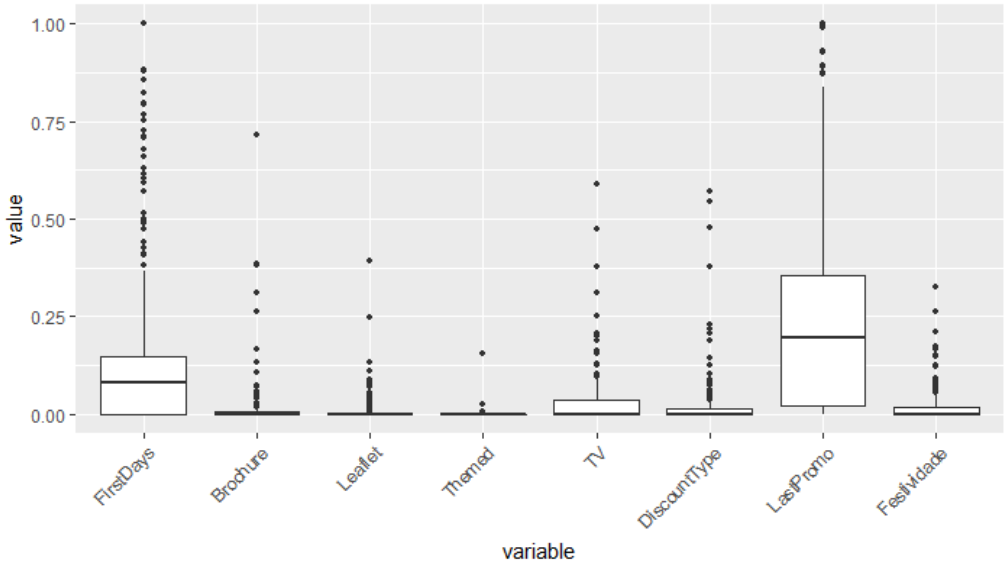
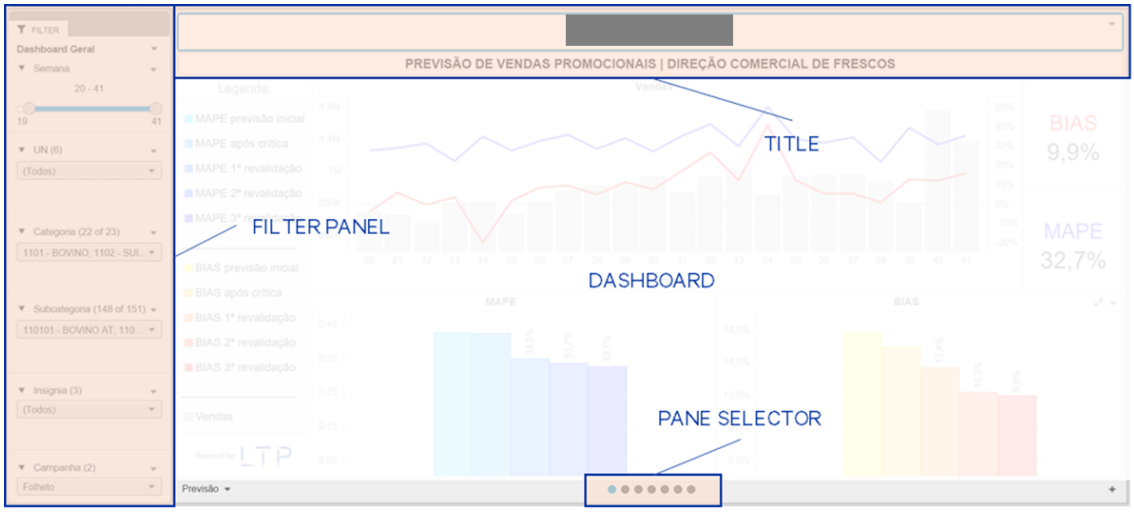



Figure C.6: Variable importance of the categorical variables in the Frozen Fish category

# Appendix D

# Dashboard

The dashboard has a skeleton that surrounds each pane and allows to filter all panes simultaneously.




1. Time frame slider (in weeks)
2. Business Unit selector
3. Category Selector
4. Subcategory Selector
5. Store Format Selector
6. Communication Means Selector

Figure D.1: Overall dashboard structure

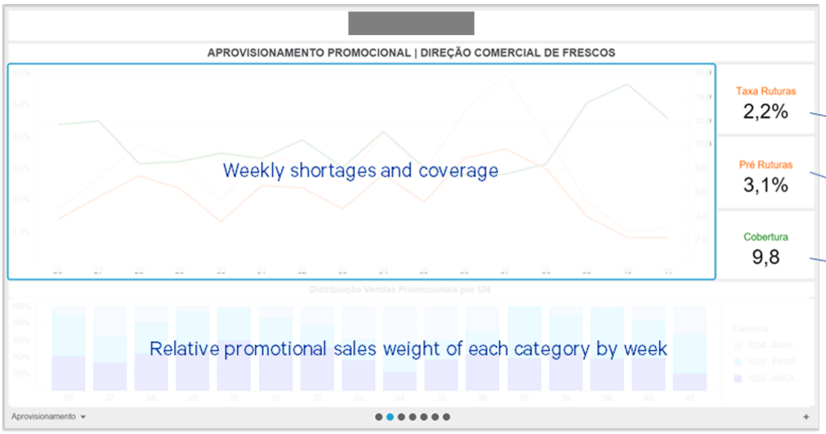




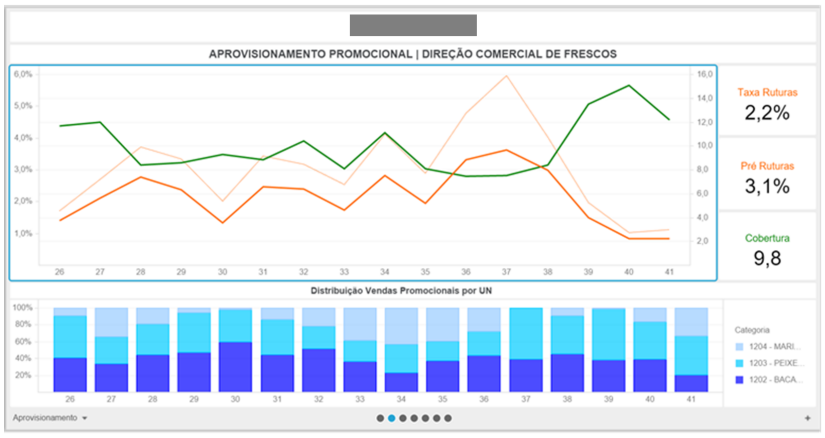
This pane aims to answer questions such as:

- Are we improving our forecasts?
- Are our predictions biased? In which direction?
- Does the store suggestion add any value? Is it being properly validated?
- Are early sales adjustments adding value?
- Were there outliers in terms of forecast error?

Figure D.2: Forecasting indicator pane



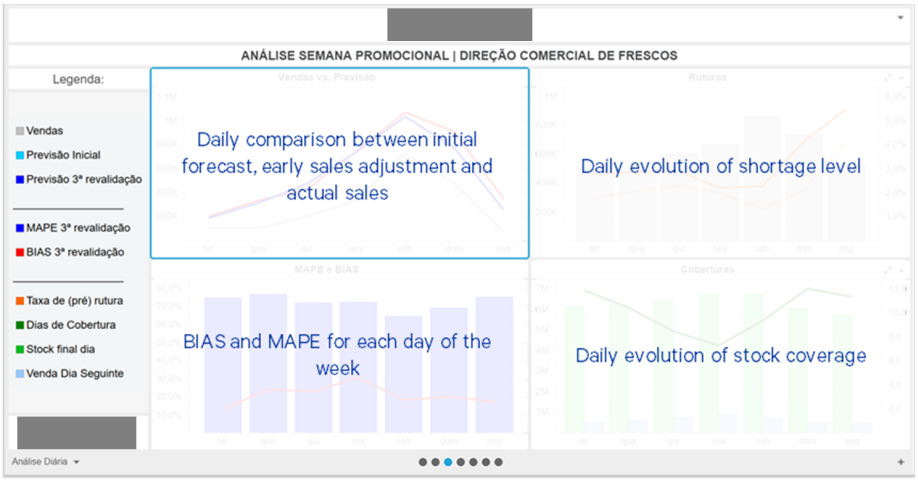

Shortages: % cases stock = 0 (weighed by average sales) | Pre-Shortage: % cases stock < 20% next day's sales
Stock coverage: # days current stock is able to cover

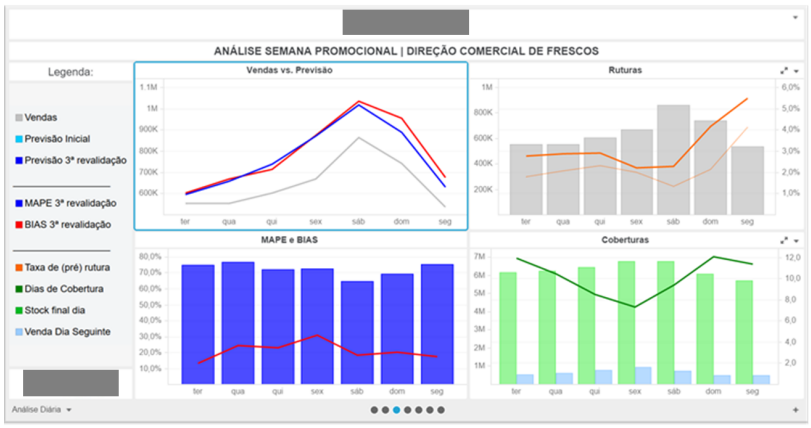



This pane aims to answer questions such as:

- Are we managing to reduce lost sales and/or coverage?
- Were there atypical weeks in terms of these KPIs?
- Does the category mix have any influence in the indicators?
- Is there a relation between shortages and coverage?
- What is the typical shortage and coverage level for this category/Business Unit?

Figure D.3: Replenishment indicator pane



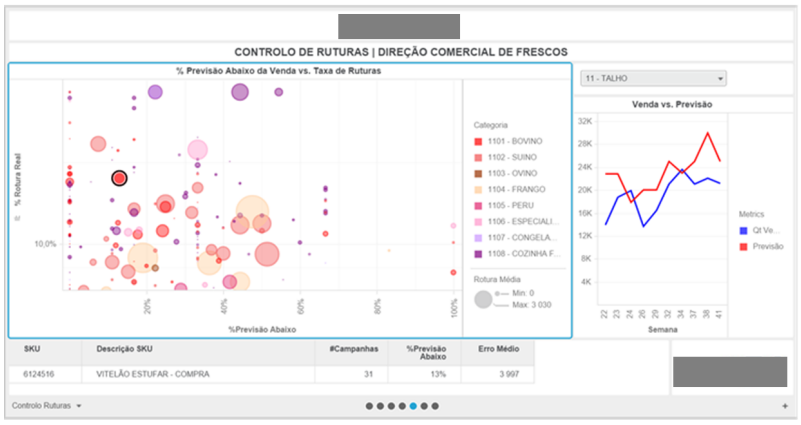




This pane aims to answer questions such as:

- What is the typical sales distribution for a week?
- In which days are shortages most common?
- Is Bias constant, or are we underestimating some days and overestimating others?
- With which stock coverage do we begin and end the week?
- Are we reacting quickly enough to early sales data?
- Are we having problems in early store filling or phase-out?

Figure D.4: Weekly indicator profile

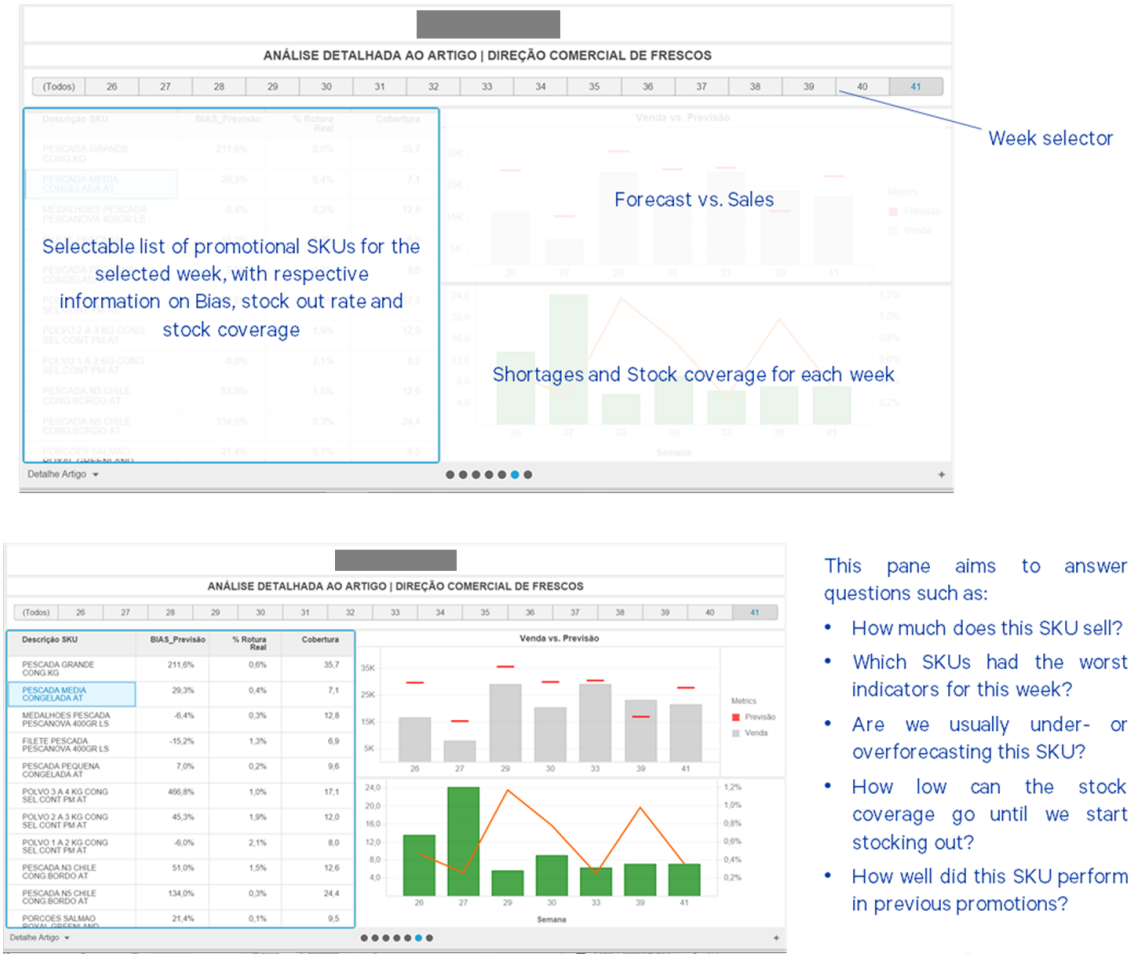

Figure D.5: Overforecasting and stock control


This pane aims to answer questions such as:

- Which SKUs are we systematically underforecasting?
- Which SKUs stock out most often?
- In which weeks have we underforecasted?
- Which are the Categories with the best/worst replenishment?
- Which SKUs must I analyse more carefully in order to effectively reduce stock out rate?

Figure D.6: Underforecasting and stock-out control



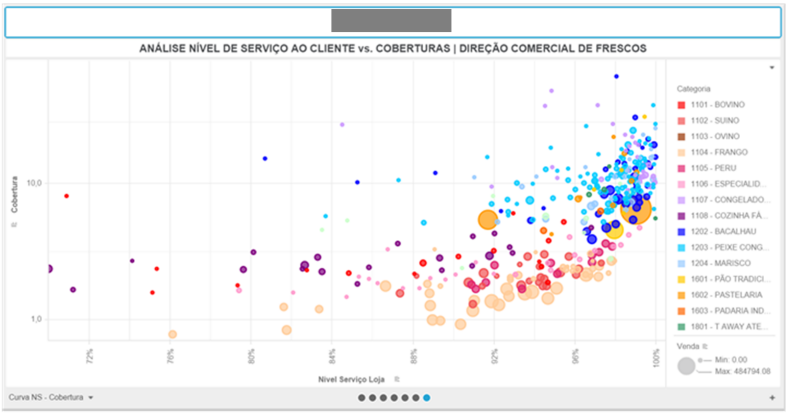




This pane aims to answer questions such as:

- How much does this SKU sell?
- Which SKUs had the worst indicators for this week?
- Are we usually under- or overforecasting this SKU?
- How low can the stock coverage go until we start stocking out?
- How well did this SKU perform in previous promotions?

Figure D.7: Detailed week view




In this pane, the Stock level vs. Service level trade-off is evident.

Here, the position of each category in the trade off can be identified by its colour. The comparative efficiency of each category is also visible, as explained in the figure below.

This pane facilitates the identification of week/category pairs that had replenishment or forecasting problems.

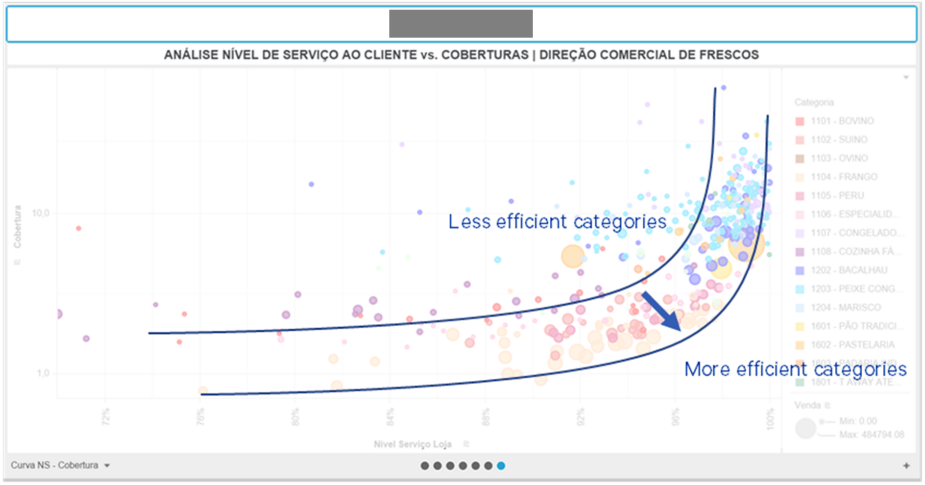



Figure D.8: The service level - stock level trade-off