UNIVERSITAT DE BARCELONA

MASTER THESIS

Optimizing Product Pricing and Sales Forecasting Through Advanced Data Science: A case Study at Schneider Electric Iberia

Author:
Jordi Segura i Pons

Directed by:
Enrique Mora Ayala
Co-supervised by:
Marina Herrera Insuela

Barcelona, June 2023

"A tothom a qui estimo i he estimat. "

"Sigo manteniendo a los que estuvieron primero."

Daniel Gómez Carrer, a.k.a Kaydy Cain

UNIVERSITAT DE BARCELONA

Abstract

MSc Fundamental Principles of Data Science

Optimizing Product Pricing and Sales Forecasting Through Advanced Data Science: A case Study at Schneider Electric Iberia

by Jordi Segura i Pons

In the increasingly competitive global business milieu, product pricing optimization and accurate sales forecasting are paramount. This MSc thesis probes these critical areas in relation to Schneider Electric Iberia, a front-runner in the digital conversion of energy management and automation. Our emphasis is on the adoption of cutting-edge data science methodologies, including econometrics, Machine Learning, causality analysis, and Deep Learning, with a goal to both predict sales and optimize price-points considering the demand elasticity of diverse products across various markets.

The thesis initiates with an in-depth analysis of Schneider Electric's extant pricing and sales forecasting systems, proceeding to the selection of suitable data science techniques for enhancement. Utilizing these methods, we devise and deploy a pricing optimization model aimed at augmenting revenue or sales volume. This model's potential is then harnessed for sales forecasting, measuring its influence on the company's operations in aspects like efficiency, profitability, and strategic decision-making amplifications.

Our methodology pivots on comprehensive data collection, meticulous preprocessing, and insightful exploratory data analysis. We leverage the benefits of Graph Causal Models for price optimization and the innovative Temporal Fusion Transformer (TFT) for sales forecasting, conjuring a formidable tool for strategic planning. The optimized prices and predictive sales model converge on an interactive Tableau dashboard, endowing Schneider Electric Iberia with a user-friendly, accessible platform for data-driven decision making. This study aims to empower Schneider Electric Iberia, while also making a noteworthy contribution to the wider field of industrial technology and the deployment of AI in product pricing and sales forecasting.

Acknowledgements

A tots vosaltres que m'heu ajudat (volgudament o no): Gràcies. Això us fa còmplices d'aquest TFM i dels meus èxits en el què em queda de vida.

Contents

Chapter 1

Introduction

1.1 Motivation

In an increasingly competitive global marketplace, the quest for efficiency and optimization permeates every corner of the business world. The realm of product pricing and sales forecasting is no exception. Here, the science of prediction and the art of optimization converge, driven by the transformative power of advanced data science methodologies. The challenge of pricing optimization and sales forecasting is a critical one for Schneider Electric Iberia. As a leader in digital transformation of energy management and automation in homes, buildings, data centers, infrastructure and industries, the ability to predict and optimize is crucial to its sustained success.

Traditionally, pricing and sales forecasting has been the purview of historical data, gut instincts and speculative market trends. However, with the advent of machine learning and artificial intelligence, this landscape has undergone a seismic shift. Advanced data science techniques such as econometrics combined with Machine Learning, causality analysis, and Deep Learning models like transformer models can now provide not only highly accurate predictions, but also the ability to optimize price points based on the elasticity of demand for different products across various markets.

The motivation behind this MSc thesis lies in the potential these advanced techniques offer. Harnessing these methodologies to find optimal pricing points that maximize revenue or sales volume has the potential to drive Schneider Electric Iberia's operational efficiency and profitability to unprecedented heights. Furthermore, the application of these techniques can provide invaluable insights into consumer behavior and market dynamics, opening new pathways for strategic decision making and competitive advantage.

Beyond Schneider Electric Iberia, the broader implications of successfully applying these techniques are vast. As Industry 4.0 and IoT continue to drive digital transformation, the efficient management of product pricing and sales forecasting will become increasingly central to the operational efficiency of industries around the world. This thesis therefore aims not only to tackle a complex problem within

Schneider Electric Iberia, but also to contribute to the wider field of industrial technology and AI application.

1.2 Objectives

This MSc thesis focuses on the application of advanced data science techniques to optimize product pricing and forecast sales. The objectives of this solution can be outlined as follows:

1. **Understanding Current Mechanisms:** The initial objective involves a meticulous investigation of the prevailing pricing and sales forecasting systems at Schneider Electric Iberia. This will involve identifying potential limitations and areas of improvement within the current strategies and methodologies.
2. **Identifying Advanced Techniques:** The next objective is to identify potent data science techniques that can be leveraged in our scenario. This will necessitate an in-depth exploration of econometrics, causality analysis, and transformer models, with a particular emphasis on their applications in pricing optimization and sales forecasting.
3. **Developing a Pricing Model:** The subsequent goal is to harness the selected techniques to devise a robust model that optimizes product pricing across various markets. The model would be calibrated to maximize either revenue or sales volume, based on Schneider Electric Iberia's strategic priorities.
4. **Implementing the Model for Sales Forecasting:** With a successful pricing model in place, our focus will then shift towards using this model to accurately forecast future sales. This will necessitate validating the model's predictive performance against historical sales data and refining it to optimize its forecasting capabilities.
5. **Assessing the Model's Impact:** The final objective is to evaluate the transformative impact of our model on Schneider Electric Iberia's operational processes. This will involve assessing improvements in efficiency, profitability, and strategic decision-making that are facilitated by our optimized pricing and sales forecasting model.

A significant portion of our efforts thus far has been dedicated to collecting, cleaning, and preprocessing data. This labor-intensive, yet fundamental step lays the groundwork for the application of advanced data science techniques, and indeed, the successful fulfillment of all our objectives.

1.3 Pipeline of the work

Our work commences with a meticulous **data collection** process explained in Chapter 3.1, striking a balance between the overarching project objectives and the requisite data to realize them. This process is far from merely aggregating figures; it involves a deep understanding of the narratives these numbers weave within the operational landscape of Schneider Electric Iberia. To achieve this, we ventured into various sources of knowledge, including the Intel Data Store (IDS), SAP, and static documentation, each contributing unique data points.

IDS, our primary resource for customer-centric information, provides detailed insights into buyer demographics, characteristics, and behaviours. The Enterprise Resource Planning (ERP) system, SAP, serves as the repository of transactional data, providing us with the historical price and sales data. Lastly, static documentation offers critical information about product changes and additional product-specific data unavailable in SAP.

Post data collection, the subsequent critical phase in our pipeline is **data preprocessing**. We initiate this with data cleaning, where incorrect, incomplete, inconsistent, and irrelevant parts of the data are identified and then corrected or removed. Next, we identify and handle missing data appropriately to maintain the integrity of our datasets.

The data then enters the **exploratory data analysis** phase explained in Chapter 3.2, where we scrutinize the collected information to understand its structures, patterns, and interactions. Here, the goal is to identify potential anomalies, formulate hypotheses for advanced statistical testing, and steer our future feature engineering and model building efforts.

Once the data is analyzed, understood, and prepared, we venture into our first significant objective: **price optimization**, well defined in Chapter 4.1. Graph Causal Models form the backbone of this phase, assisting in determining optimal price points that cater to the market's demand elasticity. This process delivers a pricing strategy that's not only competitive but also sensitive to changes in demand due to price adjustments.

Subsequently, we steer towards the second primary objective: **sales forecasting**, seen in Chapter 4.2. The Temporal Fusion Transformer (TFT) steps in here, offering a blend of automated machine learning and statistical modeling that facilitates more accurate and efficient forecasting of future sales.

Ultimately, the optimized prices are integrated with the predictive sales model. The culmination of our work is then manifested in an interactive, easy-to-understand dashboard created using Tableau that is shown in Chapter 4.3. This digital platform showcases the potential of our strategies, **visualizes** the results, and provides actionable insights for Schneider Electric Iberia's key stakeholders. All the process in this pipeline it is well appreciated in the Figure A.1.

Chapter 2

Understanding Elasticity and Forecasting

In this chapter, we delve into the foundational theories and current applications of price elasticity and sales forecasting. These concepts, vital to the effective functioning of modern economies, are significantly reshaped by the advent of advanced data science methods. Further, we explore how these concepts are pertinent to our objective of optimizing product pricing and sales forecasting at Schneider Electric Iberia. Additionally, we examine relevant literature and methodologies in these fields, illuminating their evolution and providing a context for our proposed work.

2.1 Understanding Price Elasticity of Demand

Price elasticity of demand is a fundamental concept in economics that quantifies the responsiveness of the demand for a good or service to its price changes.[1] It serves as an important measure of price sensitivity and a crucial guide in decision-making processes regarding pricing strategies.

The formal definition of price elasticity of demand (E_p) is the ratio of the percentage change in quantity demanded ($\%\Delta Q$) to the percentage change in price ($\%\Delta P$), expressed as:

$$E_p = \frac{\%\Delta Q}{\%\Delta P} \tag{2.1}$$

The significance of price elasticity lies in its ability to enable counterfactual reasoning about prices and their effects on demand. For instance, if a retailer has estimated the price elasticity of their product to be -3, they could infer that a 5% price increase would lead to a 15% decrease in demand [1].

[1] Elasticity should not be confused with slope of the demand curve. Even though both are related to the concept of responsiveness, they are different. The slope of the demand curve shows the absolute change in quantity demanded due to an absolute change in price, while elasticity shows the relative change.

With knowledge of elasticity, a retailer can make informed decisions about their pricing strategy. If the demand for a product is elastic ($|E_p| > 1$), an increase in price could lead to a substantial drop in sales, reducing total revenue. On the other hand, if the demand is inelastic ($|E_p| < 1$), demand is less sensitive to price changes and a price increase could potentially raise total revenue. When the demand is unitary elastic ($|E_p| = 1$), quantity demanded changes at the same rate as price, so total revenue remains unaffected by price changes [2], easily understood also with the representation in the Figure A.2

2.1.1 Methods of Estimating Price Elasticity

The quest for the accurate estimation of price elasticity is addressed by employing a variety of methodologies. Our work mainly leverages the Midpoint Method, Ordinary Least Squares (OLS) as an Econometric Model, Double Machine Learning (DML), and Graph Causal Models (GCM). A more comprehensive understanding of these methods can be obtained as follows:

1. **Midpoint Method:** This technique is a more refined approach to calculating elasticity, providing an average measure of responsiveness, and is especially useful when the changes in price and quantity are relatively substantial. The formula for the midpoint method is:

$$E_p = \frac{\frac{Q2-Q1}{\frac{Q1+Q2}{2}}}{\frac{P2-P1}{\frac{P1+P2}{2}}} \tag{2.2}$$

   where $P1$ and $P2$ represent the initial and final prices, and $Q1$ and $Q2$ represent the initial and final quantities. By calculating the percentage change relative to the midpoint rather than the initial value, the midpoint method offers a more balanced measure of elasticity [1].

2. **Ordinary Least Squares (OLS):** OLS is a cornerstone method in econometrics, commonly used in the estimation of price elasticity. It minimizes the sum of the squares of the residuals, i.e., the differences between the observed and predicted values. The formula of the OLS estimator in a simple linear regression model is:

$$\hat{\beta} = (X'X)^{-1}X'Y \tag{2.3}$$

   where $\hat{\beta}$ represents the OLS estimator, X is the matrix of independent variables, Y is the dependent variable vector, and $'$ denotes matrix transposition. Although OLS models are simple and interpretable, they require careful specification and understanding of potential confounding factors [3].

3. **Double Machine Learning (DML):** This advanced method combines machine learning and econometrics to control for confounding variables when estimating price elasticity. The DML procedure typically involves the following steps:
   (a) **De-meaning Process**: Adjusting for baseline variations in price and quantity per product type by subtracting their mean values.
   (b) **Feature Generation**: Generating a set of features using potential covariates like date, product type, month, and customer class.
   (c) **Model Training**: Training two Random Forest Regressors, one for predicting quantity and the other for predicting price.
   (d) **Prediction and Residualization**: Predicting the log of quantity and price using the trained regressors and computing residuals.
   (e) **Elasticity Estimation**: Estimating price elasticity by fitting an Ordinary Least Squares (OLS) regression model on the residuals.

   The use of machine learning to control for confounders increases the accuracy of elasticity estimates, particularly in complex scenarios with high-dimensional covariates [4].

4. **Graph Causal Models (GCM):** GCM, grounded in graph theory, visually illustrates the relationships between different variables. This visualization aids in comprehending and managing the complex causal relationships inherent in price elasticity scenarios. GCM can capture and analyze the multifaceted interactions between numerous variables, enabling more accurate and informed sales forecasting.

2.1.2 Causality and Elasticity

Gaining an in-depth comprehension of price elasticity necessitates understanding the notion of causality, which addresses the cause-and-effect associations between different variables or events [5]. In the context of price elasticity, we focus on the causal link between price, acting as the cause, and demand, being the effect.

An essential element of this causal examination is the application of counterfactuals, which facilitate envisioning alternative realities to actual events. This concept allows us to consider questions such as, "How would the demand be impacted if we reduced the price by 5%?"

Nevertheless, causality often transcends a straightforward cause-and-effect relationship and is usually interspersed with confounding factors – variables that concurrently influence both the cause and the effect. To illustrate, within our price elasticity scenario, the product's quality could serve as a confounder. For instance, a high-quality, expensive product like a MacBook may sell more units than a lower-priced, lower-quality product like a Chromebook. This could mislead us into believing that a higher price escalates demand. However, it's the quality of the product that simultaneously affects both the price and demand (See Figure A.3).

Modern advancements in data science and econometrics allow us to overcome these challenges, enabling us to assess price elasticity with enhanced precision, even in the presence of confounding variables. These advancements permit the application of econometric models and machine learning techniques to scrutinize granular sales data and guide informed pricing decisions [6]. Hence, the criticality of a comprehensive understanding of the price elasticity of demand and its estimation methods for maximizing revenue in diverse market conditions cannot be overstated.

The acknowledgment of these concepts is pivotal for accurate price elasticity estimation. Overlooking the role of confounding factors may introduce bias in our elasticity estimates, potentially misleading us in formulating optimal pricing strategies [7]. Therefore, state-of-the-art econometric models and advanced machine learning methodologies strive to account for these confounders, leading to precise elasticity estimates.

Graph Causal Models

Graph Causal Models (GCMs) are a powerful framework grounded in graph theory that enables us to analyze complex causal relationships among variables. GCMs provide a visual representation of the causal structure, allowing us to gain insights into the intricate web of causal dependencies inherent in price elasticity scenarios. By leveraging the principles of graph theory, GCMs facilitate a deeper understanding of the causal relationships between pricing and sales dynamics, aiding us in the development of accurate and reliable sales forecasting and price optimization strategies.

In GCMs, variables are represented as nodes in a graph, and causal relationships are depicted as edges connecting the nodes. This graphical representation helps us visualize and comprehend the complex interactions and interdependencies among different variables. GCMs enable us to identify direct causal effects and indirect effects mediated through intermediate variables, capturing the multifaceted nature of the causal relationships. The analysis of GCMs allows us to investigate the direct and indirect impacts of pricing on sales, identifying the key drivers and uncovering the underlying causal structure.

Formally, we can represent a GCM as a directed acyclic graph (DAG), where each node corresponds to a variable, and the edges indicate causal relationships. Let G = (V, E) denote our GCM, where V represents the set of variables and E represents the set of directed edges. For each edge $(i, j) \in E$, it signifies a causal relationship from variable i to variable j. By analyzing the graph structure and considering the directionality of the edges, we can infer the causal dependencies among the variables and discern the factors that influence sales, see figure A.3.

To quantify the causal relationships and estimate the effects of variables on sales, we can employ various causal inference methods within the GCM framework. These methods include but are not limited to structural equation modeling, Bayesian networks, and potential outcome frameworks such as counterfactual analysis. By utilizing these methods, we can estimate the causal effects of pricing decisions on sales while accounting for confounding variables and other potential biases.

The application of GCMs in demand forecasting and price optimization provides several benefits. Firstly, GCMs enable us to identify confounding variables, which are factors that simultaneously influence both pricing and sales. By explicitly modeling and controlling for these confounders, GCMs help us disentangle the direct causal effects of pricing from the indirect effects mediated through other variables. This facilitates a more accurate estimation of the causal impact of pricing decisions on sales, enabling us to make informed pricing strategies and optimize revenue generation. [8]

Secondly, GCMs enhance the interpretability of the relationships between pricing and sales. The graphical representation of the causal structure allows us to visually examine and understand the complex causal pathways and interdependencies among variables. This interpretability aids us in identifying key drivers of sales and provides insights into the underlying mechanisms governing the pricing-sales relationship.

Furthermore, GCMs can incorporate additional contextual information and external factors that may influence sales. By including exogenous variables or external factors as nodes in the graph, GCMs allow for a comprehensive analysis of the impact of various factors on sales. This holistic view enables us to develop robust forecasting models that account for a wide range of influencing factors, leading to more accurate and reliable predictions.

By utilizing GCMs, we can identify potential confounding variables, gain insights into the direct and indirect effects of pricing on sales, and ultimately, generate more accurate and reliable sales forecasts [9].

Particularly for scenarios such as ours, where price and demand dynamics are influenced by a multitude of interconnected factors, the application of GCMs is instrumental in revealing the underlying causal structure, thereby enhancing the precision and reliability of sales forecasting and price optimization.

The incorporation of these methodologies is instrumental in estimating price elasticity with high accuracy, accommodating the intricacies of real-world data. Understanding these methods is crucial for interpreting the work presented in this thesis.

In the next section, we will probe into sales forecasting, exploring its significance, methodologies, and its connection with our research on price elasticity.

2.2 Sales Forecasting

Sales forecasting represents a critical process of predicting future sales volumes, acting as a pivotal driver for companies' business decisions and offering visibility into both short-term and long-term performance trajectories. As a key component of strategic management, accurate sales forecasts enable organizations to effectively plan production schedules, manage resources, and control financial activities, fostering operational efficiency [10].

In recent years, the realm of sales forecasting has been radically transformed by the advent of data science methodologies, particularly machine learning and artificial intelligence technologies. These innovative methodologies present a substantial departure from traditional statistical approaches, offering the capacity to navigate complex data patterns and manage large-scale data sets. As a result, they yield significantly enhanced forecast accuracy [11].

A case in point is the emergence of transformer models, a specialized category of deep learning models that have demonstrated exceptional proficiency in capturing long-range dependencies within data. This capability positions them as an ideal fit for sales forecasting applications, where understanding historical trends and their influence on future outcomes is crucial [12].

Enriching our sales forecasting model with the optimized pricing derived from our elasticity analysis, we aim to augment the precision of our sales predictions. This integration allows the model to account for the expected shifts in demand ensuing from the proposed price adjustments, thereby creating a more holistic, responsive, and accurate forecasting mechanism

2.2.1 Traditional vs. Modern Approaches

Traditional sales forecasting methods, commonly utilized for decades, are grounded in statistical analysis. Time series analysis, including moving average models, autoregressive models, and exponential smoothing, has historically been employed extensively.

Time series models focus on historical data, using past sales patterns to forecast future sales. These models assume that the future will be a function of the past, which makes them simple and effective when dealing with stable and linear trends. However, they often struggle when facing more complex scenarios. For instance, if market dynamics undergo significant changes or if non-linear patterns exist in the sales data, traditional models' performance may deteriorate.

One common method is the *Autoregressive Integrated Moving Average* (ARIMA) model, which takes into account three aspects: autoregression (the relationship between an observation and a number of lagged observations), differencing (making the time series stationary), and moving average (the dependency between an observation and a residual error from a moving average model applied to lagged observations).

Exponential smoothing is another classic forecasting method. This approach applies weights that decrease exponentially. The most recent observation has the highest weight, and the weights decrease for older observations. The popular *Holt-Winters* method, an extension of exponential smoothing, considers seasonality in the data.

Modern sales forecasting methods, propelled by advancements in machine learning and artificial intelligence, are capable of dealing with the shortcomings of traditional approaches. They offer superior performance in handling non-linear and complex data patterns, making them ideal for today's volatile and rapidly changing markets.

Machine learning-based forecasting methods encompass a wide range of techniques. These include linear regression models, tree-based models like Random Forest and Gradient Boosting Machines (GBM), Support Vector Machines (SVM), and neural networks.

Linear regression models, while simple, can be powerful tools when relationships between variables are linear. But they may fail when dealing with complex, non-linear relationships in data.

Tree-based models, such as Random Forest and GBM, are non-parametric and can handle both linear and non-linear relationships. They also allow interactions between variables. However, these models can become overly complex, leading to overfitting where the model learns the training data too well and performs poorly on new data.

On the other hand, deep learning models, including Recurrent Neural Networks (RNN), Long Short-Term Memory (LSTM) units, and Gated Recurrent Units (GRU), have shown great promise in sales forecasting. These models are excellent for sequential data, as they capture long-term dependencies in the data. This makes them ideal for time series forecasting.

Transformative methodologies like the transformer model represent the latest advancements in this space. The transformer model, with its attention mechanism, allows the model to focus on different parts of the input sequence when making predictions, enhancing the accuracy of sales forecasts.

2.2.2 Temporal Fusion Transformer: Model Architecture

The Temporal Fusion Transformer (TFT) is a powerful, versatile, and highly interpretable model developed for time series forecasting. It's specifically designed to accommodate different types of inputs, including static, known past, and observed past inputs, and employs specialized networks and encoders to generate high-performance forecasts.

TFT incorporates a range of components to maximize its effectiveness and adaptability, all of them can be observed in the Figure 2.1 where the architecture is represented. These include gating mechanisms, which allow the model to adapt to a broad spectrum of datasets by bypassing unused components, and variable selection networks that improve generalization by prioritizing the most salient features. Additionally, the TFT uses static covariate encoders to incorporate static features into the modeling of temporal dynamics, thereby ensuring that the impact of static factors, such as a store location, is adequately accounted for in the forecasts.

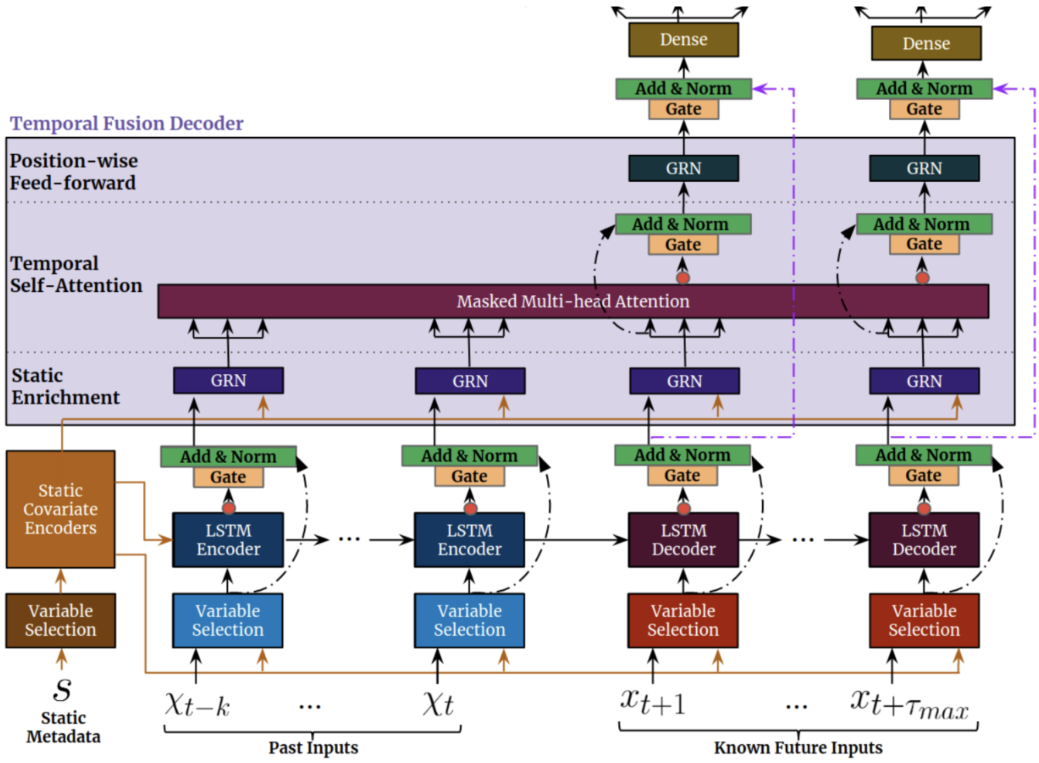


FIGURE 2.1: Bryan Lim et al, 2020. [13]

Gating Mechanisms

The TFT utilizes Gated Residual Networks (GRNs) as building blocks. These mechanisms allow the model to adapt its depth and network complexity by selectively bypassing unused components, providing flexibility and accommodating a wide range of datasets and scenarios.

The GRN takes in a primary input a and an optional context vector c and yields the output $GRN_\omega(a, c)$ as follows:

$$\text{GRN}\omega(a,c) = \text{LayerNorm}(a + \text{GLU}\omega(\eta_1)) \tag{2.4}$$

where $\eta_1 = W_{1,\omega}\eta_2 + b_{1,\omega}$ and $\eta_2 = \text{ELU}(W_{2,\omega}a + W_{3,\omega}c + b_{2,\omega})$.

Here, ELU represents the Exponential Linear Unit activation function, η_1 and η_2 are intermediate layers, LayerNorm is the standard layer normalization, and ω is an index denoting weight sharing.

The Gated Linear Units (GLUs) are used as component gating layers, which provide the flexibility to suppress any parts of the architecture that are not required for a given dataset. The GLU takes the following form:

$$\text{GLU}\omega(\gamma) = \sigma(W4,\omega\gamma + b_{4,\omega}) \odot (W_{5,\omega}\gamma + b_{5,\omega}) \tag{2.5}$$

where $\sigma(\cdot)$ is the sigmoid activation function, $W_{(.)}$ and $b_{(.)}$ are the weights and biases, and $\odot$ represents the element-wise Hadamard product.

By combining the GRNs and GLUs, the TFT can control the extent to which the GRN contributes to the original input a, potentially skipping over the layer entirely if necessary by setting the GLU outputs close to 0.

During training, dropout is applied before the gating layer and layer normalization to η_1.

The flexibility provided by these gating mechanisms allows the TFT to adapt its network architecture based on the characteristics of the dataset, enabling it to effectively handle a wide range of scenarios and achieve high performance in time series forecasting tasks.

Variable Selection Networks

The variable selection networks ascertain the relevance of input variables at each time step, thereby focusing on pivotal features and mitigating the influence of noisy inputs. This enhances model performance by utilizing the learning capacity optimally on the most informative variables.

In the TFT, the variable selection networks scrutinize both static and time-varying covariates to discern the most critical variables for prediction, shedding light on those with the highest predictive content. Categorical variables are represented using entity embeddings, while continuous variables undergo linear transformations for compatibility with subsequent layer dimensions.

Implemented separately for static and time-varying covariates, the variable selection networks work on similar principles. Here, the focus is on the time-dependent covariate selection, noting that analogous principles apply to static covariates.

The transformed input of each variable ($\xi_t^{(j)}$) is processed through the variable selection network at every time step. The output, variable selection weights ($v_{\chi t}$), signifies the importance assigned to each variable.

These weights are derived by passing the transformed inputs (Ξ_t) and an external context vector (c_s) through a Gated Residual Network (GRN). The GRN considers both inputs to generate the variable selection weights.

These weights dictate the contribution of each variable to the final prediction per time step. TFT, by incorporating these weights, selectively accentuates the most significant variables while minimizing the influence of the less informative ones. Thus, variable selection networks are critical in tailoring TFT's learning focus for improved forecasting.

Encoder and Decoder

The TFT (Temporal Fusion Transformer) incorporates static metadata by utilizing separate GRN (Gated Residual Network) encoders to produce context vectors. These context vectors play a crucial role in incorporating static features into the temporal dynamics of the model. By utilizing separate encoders, TFT ensures that the impact of static factors, such as static covariates or metadata, is adequately accounted for in the forecasts.

The temporal fusion decoder in TFT is responsible for learning temporal relationships within the time series data. It consists of several layers that work together to capture dependencies and enhance forecasting performance.

1. **Sequence-to-Sequence Layer**: The decoder begins with a sequence-to-sequence layer, which is responsible for enhancing the locality of the input data. This layer leverages local context and patterns to improve forecasting accuracy. In TFT, a sequence-to-sequence model, such as an LSTM encoder-decoder, is used to handle the differing number of past and future inputs.

2. **Static Enrichment Layer**: Following the sequence-to-sequence layer, TFT incorporates a static enrichment layer. This layer enhances the temporal features with static metadata. The static enrichment layer ensures that static covariates and metadata have a significant influence on the temporal dynamics of the model.

3. **Temporal Self-Attention Layer**: TFT utilizes a temporal self-attention layer to capture dependencies and relationships across different time steps. The self-attention mechanism allows the model to learn long-range dependencies and identify important patterns in the time series data. Masking techniques are applied to preserve causal information flow within the self-attention layer.

4. **Position-Wise Feed-Forward Layer**: After the temporal self-attention layer, TFT applies a position-wise feed-forward layer. This layer introduces non-linear processing to the outputs of the self-attention layer, further enhancing the representation and capturing complex temporal relationships.

Gating mechanisms and skip connections are employed throughout the encoder and decoder to facilitate training and simplify the model's complexity when additional complexity is not required. These mechanisms provide adaptive depth and network complexity to accommodate a wide range of datasets and scenarios.

By integrating static metadata through separate encoders and utilizing the various layers in the decoder, TFT effectively learns and models the temporal dynamics of the time series data, capturing dependencies, incorporating static factors, and producing accurate forecasts.

Multi-Head Attention Mechanism

The Temporal Fusion Transformer (TFT) utilizes a self-attention mechanism, a refined form of the multi-head attention employed in transformer architectures. This mechanism enables TFT to identify long-term dependencies across various time steps in the input sequence.

Attention mechanisms typically assign weights to values based on the relationships between keys and queries. The attention function, noted as Attention(Q, K, V), calculates weighted sums of values influenced by the similarity between keys and queries. Scaled dot-product attention is a popular choice where attention weights are derived from applying a softmax function to the dot product of queries and keys, normalized by the square root of the query dimension.

To augment the standard attention mechanism's learning capacity, multi-head attention is incorporated. Here, different attention heads operate in different representation subspaces, each with its unique key, query, and value weights. The outputs from all heads are concatenated and linearly combined using head-specific weights, forming the final representation.

TFT modifies the multi-head attention for enhanced explainability and interpretability. Contrary to using distinct values for each head, TFT shares the values across all heads. Consequently, attention weights alone don't denote a specific feature's importance. The outputs of all heads are then aggregated additively, resulting in a joint representation that detects multiple temporal patterns and pays attention to a shared set of input features.

This altered multi-head attention in TFT, known as "interpretable multi-head attention," allows the model to discern diverse temporal patterns while focusing on pertinent features in an interpretable, efficient manner. The value-sharing across heads facilitates a collective ensemble over attention weights, thus amplifying representation capacity.

Quantile Outputs

In addition to point forecasts, the TFT model also generates prediction intervals or quantile forecasts. This allows for a probabilistic representation of the forecasted values and provides an estimate of the uncertainty associated with the predictions.

To generate quantile forecasts, the TFT model simultaneously predicts various percentiles at each time step, such as the 10th, 50th, and 90th percentiles. Each quantile represents a specific level of confidence or probability. The quantile forecasts are obtained through a linear transformation of the output from the temporal fusion decoder.

The formula for calculating the quantile forecast at a given quantile q, time step t, and prediction horizon τ is as follows:

$$\hat{y}(q, t, \tau) = W_q \tilde{\psi}(t, \tau) + b_q \tag{2.6}$$

Here, W_q represents a linear coefficient matrix of size $1 \times d$, and b_q is a bias term of size 1. The $\tilde{\psi}(t, \tau)$ term refers to the output from the temporal fusion decoder at time step t and prediction horizon τ. By applying the linear transformation, the TFT model produces the quantile forecast for the specified quantile at each time step.

It's important to note that the forecasts are only generated for future horizons, denoted by $\tau \in 1, \ldots, \tau_{\max}$. This ensures that the model focuses on predicting values beyond the current time step.

By generating quantile forecasts, the TFT model provides not only point estimates but also a range of likely values at different confidence levels. This enables decision-makers to assess the uncertainty associated with the forecasts and make informed decisions based on the probabilistic nature of the predictions.

2.3 Related Works and State-of-the-art

The field of demand forecasting and optimization has seen an impressive evolution over the years with the advent of AI technologies. From simpler statistical methods to complex machine learning models, and more recently the rise of advanced deep learning methods, the state-of-the-art has been continually advancing. It is worthwhile to note that, while the simpler techniques have been largely superseded, they still form a fundamental building block for understanding the core concepts and intuition behind more sophisticated methods.

Historically, various models have been used for demand forecasting and optimization tasks, from traditional time series models such as Autoregressive Integrated Moving Average (ARIMA) [14], Exponential Smoothing (ETS), and state space models, to more modern machine learning methods, such as Random Forests and Support Vector Machines. These methods have shown good performance in specific scenarios, but often suffer when the forecasting problem involves complex temporal dependencies, multivariate inputs, or when the data is affected by non-linear relationships and changing dynamics [15].

In recent years, deep learning methods have been increasingly adopted in demand forecasting and optimization tasks, given their ability to model complex, non-linear relationships, and their capacity to capture long-term temporal dependencies. Recurrent Neural Networks (RNNs), and particularly their variant Long Short-Term Memory (LSTM), have been extensively applied for demand forecasting, due to their intrinsic ability to handle sequential data. However, LSTMs often struggle when dealing with very long sequences and multiple temporal patterns [16]. This has prompted research towards models that can capture various temporal patterns and hierarchies, leading to the development of models such as the Temporal Fusion Transformer (TFT).

The TFT, introduced by Bryan Lim et al. in 2020, represents a significant milestone in the evolution of demand forecasting models. By using a mix of convolutional and recurrent layers, attention mechanisms, and gating, the TFT is capable of modeling complex temporal relationships, variable selection, and interpretability. The authors demonstrated that the TFT outperforms several baselines on a range of forecasting tasks, thus establishing it as a state-of-the-art model for demand forecasting [13].

Parallel to the developments in demand forecasting, the field of optimization has seen considerable advances. Traditional optimization techniques such as Linear Programming (LP) and Integer Programming (IP) have been widely used in various applications. However, these methods often struggle with problems that involve complex constraints, non-linear objectives, or uncertainty in the parameters.

In the realm of AI, Reinforcement Learning (RL) has emerged as a powerful tool for solving complex optimization problems. More recently, the concept of Graph Neural Networks (GNNs) has been introduced for optimization tasks on graph-structured data. The work of Joshi et al. [17] exemplify this approach, demonstrating the efficiency of GNNs in solving classical optimization problems like the Travelling Salesman Problem (TSP) and the Vehicle Routing Problem (VRP).

Notably, the introduction of Graph Causal Model (GCM) by Peng et al. [18] marked an important advancement in the intersection of optimization and causal inference. GCMs exploit the structure of causal graphs to improve optimization in scenarios with inherent causal structures, similar to what Narendra et al. [19] proposed in their work where they used counterfactual reasoning for optimization using Causal Models .

Our work builds on this concept, further extending it to the intersection of demand forecasting and optimization. We aim to leverage the power of causal inference within the context of demand forecasting to enhance optimization strategies. By incorporating causal graphs into our framework, we can capture the inherent causal relationships between various factors influencing demand and utilize this knowledge to improve the accuracy and efficiency of forecasting and optimization processes.

Inspired by the success of deep learning models like the TFT in demand forecasting, we propose a novel approach that combines the strengths of causal inference and advanced machine learning techniques. Our framework integrates Graph Causal Models (GCMs) with state-of-the-art deep learning architectures to create a unified solution for demand forecasting and optimization problems.

Chapter 3

Data Acquisition, Preprocessing and Exploratory Data Analysis

The heart of our endeavor lies within the depths of Schneider Electric Iberia's data. It is here that the potential for uncovering optimal pricing points and forecasting sales volumes resides. The third chapter of our journey, therefore, turns to the vital stages of the Acquisition, Data Preprocessing and Exploratory Data Analysis (EDA). These phases are not merely preliminary steps, but they form the backbone of our investigation, laying the groundwork for the powerful methodologies we will employ.

The raw data that emerges from Schneider Electric Iberia's diverse markets and myriad product lines is both a treasure trove of insights and a labyrinth of complexity. To navigate this labyrinth and extract the hidden treasure, we must diligently clean, structure, and explore our data. It is through these careful preparations that we ensure the subsequent advanced data science techniques can truly shine and deliver the insights we seek.

3.1 Crafting Our Dataset

Unraveling the intricacies of our data is akin to navigating a complex labyrinth. This necessitates a granular dissection of each information source we leverage, revealing the unique utility each one offers.

Our first tryst with the data begins with SAP, our primary information repository. From its vast troves, we extracted price references for all materials in one specific Helios[1], the HDPNL. This Helios, as one of the primary product lines in Iberia, holds immense potential for insightful analysis and impactful results. It is important to

[1] Helios, in the context of Schneider Electric, refers to a classification system for products, serving as a method of organization and categorization. It allows for more efficient management and understanding of the diverse range of products.

note that this work it can be replicated to others Helios, but we did not do it for both time and computationally resources.

However, deciphering this dataset is not as straightforward as it may seem. The price data does not follow a monthly organization but instead provides a range of dates signifying when prices have changed for each material. These changes, usually incremental, span over natural years from February of 2017 through to April of 2023. The journey of deciphering this dataset is arduous, but it's an endeavor that will undoubtedly yield significant rewards. Our voyage through the data was not devoid of challenges. As we delved into it, we encountered anomalies such as erroneous prices like 0.00 and 9999.99. These anomalies, representative of incorrect prices or special offer materials, were meticulously weeded out to maintain the integrity of our data.

Having cleaned the dataset, we amalgamated the information spanning over multiple years into a single, cohesive Dataframe. Each material was assigned a unique price per month, derived from the date ranges mentioned earlier. Progressing further, we leveraged the static file received from the business to update outdated references, imbuing our data with additional information on Activity, Subactivity, and Status for each material. This enrichment was essential, particularly the Status data, to account for materials that have been discontinued or are no longer active.

At the end of this meticulous exercise, we were left with a significantly refined dataset of 217,289 rows, down from the initial 635,856 rows of activity. This translates to over a 60% reduction through our cleansing process, demonstrating our commitment to ensuring the precision and relevance of our data.

As our journey continued, we shifted our focus to the sales information, embarking once more into the depths of SAP. This time, we targeted historical data from 2017 through 2023 for the HDPNL Helios, amounting to nearly 3 million rows of data. It's worth noting that our sales dynamics are anything but simple. While the price of a sale tends to be based on the Material, disregarding the identity of the buyer or the channel of distribution, the landscape changes when we delve into sales data. Here, we are faced not with standardized prices but final prices, teeming with potential discounts, special offers, and customizations particular to each transaction. Hence, it's vital to consider certain crucial elements when examining the sales dataset we collected:

- **Account Sold To ID:** This pertains to the ID of the client to whom the material was sold. We chose not to delve into this level of granularity for the current work.
- **Quantity:** This signifies the number of units sold in a transaction.
- **Gross Sales:** This represents the total revenue garnered from the transaction. It's crucial to remember that this is not typically the product of the quantity

and the price; rather, it may reflect a discounted price, the details of which are not explicitly included in the dataset.

- **Type of Transaction:** The nature of a transaction can provide valuable insights regarding the quantity sold or any discount applied. For our work, we focused mainly on three types: **ZCAM, ZOES, and ZNOR**.
- **Sales Territory**: An information of the territory in Iberia we did the transaction, which might give some valuable details to the models.

Having meticulously integrated the sales data from different years, we proceeded with several essential data cleaning steps. This included filtering out Materials that did not appear in at least three different years and six months per year, or those that had sales below zero due to specific transaction types. Simultaneously, we engineered new features such as the real price and the percentage change between months. The next logical stride in our data journey was to fuse the price and sales dataframes.

After successfully merging the datasets, we incorporated insights from our CRM data stored in Intel Data Store, extracting vital data concerning the type of customer for each sale (B2B, B2C, Design Firm, and so forth). Equipped with this enriched dataset, we set our sights on an in-depth Exploratory Data Analysis (EDA) and Feature Engineering, seeking to uncover actionable insights for our subsequent steps. It's noteworthy that at this juncture, our dataset had expanded to nearly 2.5 million rows, providing us with an abundance of data for our in-depth examination.

3.2 Exploratory Data Analysis

This section combines advanced visualization tools and data processing techniques to understand our data's complex landscape and translate these insights into comprehensible information for our business stakeholders. We'll complement visual explorations with in-notebook analysis to highlight less apparent aspects of our data and eliminate unessential elements. In essence, this journey involves a strategic blend of visualization and analysis, jointly charting our course through Schneider Electric Iberia's intricate data landscape.

We embark on our Exploratory Data Analysis by detailing our utilization of Tableau. A glance at Figure A.4 reveals the structure of our dashboard. At the top, the selected Helios and the count of materials within it are distinctly visible, dynamically responding to applied filters. These filters, located on the top-right, offer customization based on Subactivity—our grouping methodology for materials—and Transaction Type, which, as discussed earlier, plays a significant role in pricing variances across types.

A key metric presented here is the Schneider-referred 'Precio medio' or the 'final selling price'. This figure, more insightful than the standard list price, denotes the final price at which a product is sold. The box plot at the top illustrates this for selected subactivities (primarily those from Top References). We observe that the majority of prices fall within reasonable levels, while a small fraction veer towards the higher end. Nevertheless, these outliers do not present a significant cause for alarm.

Beneath this, we've implemented a packed bubble graph displaying a cloud of prices—an instrumental visual tool for spotting outliers and understanding how high-priced materials cluster. The size and color of the bubbles represent the final price and their grouping, respectively. To its right, another packed bubble graph represents the distribution of material prices, categorized into bins of hundreds. While this visualization is more business-focused than data-focused, it provides a quick overview of our final price distribution, a crucial aspect which was not explicitly represented in our data and lacked visualization tools in our business repertoire.

The top right quadrant showcases two distinct boxplots, each offering insights into Transaction Type and Customer Classification, as seen in Figure A.5. These graphics readily illustrate the variations in final prices across different Transaction Types, a contrast less prevalent in Customer Classifications. Here, it's worth noting that B2B and B2C final prices appear strikingly similar, with more significant discrepancies present in other categories like End Users.

In the lower half of the dashboard, as depicted in Figure A.6, we find a boxplot that underscores the cumulative percentage change in prices from 2017 to 2023. This particular perspective offers a unique understanding of the data, not in terms of individual prices or final prices, but in the effective evolution of these prices over time.

Further augmenting this perspective, the two packed bubble graphs below trace the behavior of these prices. The homogeneity of color illustrates a consistency in price increment patterns, which is logical considering price adjustments typically span entire subactivities rather than individual materials.

The graph on the bottom right introduces another vital aspect for our future analyses—the rate of price increase across these subactivities. It's worth noting that price increases do not necessarily mirror the hikes in final prices. Superimposed on this graph is a histogram, tallying the quantity of materials present over the years. It shows a stable trend for these categories, an expected finding given these represent the most significant subactivities within our Helios.

Delving into the depths of our data, we enriched it with insights acquired through advanced visualizations. A realization emerged: not all materials bear equal significance, and similarly, not all subactivities carry the same weight. Informed by these insights and our continued dialogue with the Head of Pricing in Iberia and the VicePresident of Data, we crafted a couple of plots showcasing these disparities.

The first plot (Figure A.7) illuminates the prominence of each subactivity, and the second (Figure A.8) highlights the importance of each material. We quantified 'importance' via a metric referred to as 'weights' by the business. Weights represented the proportion of quantity sold within the entirety of Helios when evaluating subactivities, and within the specific subactivity when evaluating materials.

The evolution of subactivities (Figure A.7) paints an intriguing picture. Back in 2017, a mere seven subactivities constituted over 80% of the Helios' total sales. This trend persisted through 2022, save for one subactivity which ceased to contribute to sales. Intriguingly, this pattern mirrored Pareto's principle: 20% of all subactivities within Helios (6 out of 30) accounted for 80% of the total sales. This data-driven insight, coupled with business logic, reinforced our decision to focus our analysis on these six prime subactivities.

Looking within these subactivities, we observed a similar pattern with materials. As depicted in Figure A.8, a long tail of less significant materials existed alongside a select few dominant ones. The solution lay in weighting each material within its subactivity and retaining only the top 3.
As shown in our results (Figure 3.1), these top 3 materials constituted nearly 20% of sales for their subactivity - a considerable share for just 3 references. Furthermore, aligning our data-driven insights with business acumen, we considered the business-identified 'Referencias Faro'. These critical materials play a pivotal role when it comes to price changes and adjustments.

In conjunction with our exploratory data analysis, and leveraging our cleaned dataset, we aimed to augment the business value by generating more nuanced insights. Accordingly, we devised two additional dashboards.

The initial dashboard shown in Figure A.9 dissects the sales data. The topmost chart illustrates the progressive increase in the Schneider Electric (SE) price for select subactivities, adjusted using the provided filters. Notably, we present the data at two levels of granularity—subactivity and material, enhancing the precision of our analysis.

The subsequent chart highlights an interesting trend—while the SE price has escalated, the final price has remained relatively stable over the past five years. This suggests that despite the rise in price, our commercial team has maintained or even augmented the final discounts offered to sustain sales. This trend, however, is somewhat offset by the positive news that the volume for these subactivities has remained steady, indicating stable demand. The last pair of charts break down the final price and quantity sold for these aggregated subactivities by transaction type. It is evident from these visualizations that transaction types ZCAM and ZOES hold a majority share in sales.

The second dashboard in Figure A.10 provides comparable insights with a focus on customer-related data. The initial three charts delineate the trajectory of the final price for different subactivities and customer types, the prominent increase in volume for our B2B clientele, and the customer purchasing trends in a typical month.

These graphical representations provide insights into pricing strategies for different customer types and subactivities, volume trends for B2B customers, and the usual buying patterns of various customers within a month.

The last set of three histograms delivers a ranked classification showcasing our highest buying customers, the customers with the highest purchasing frequency, and those contributing significantly to our revenue.

After this Exploration Data Anlysis phase, we decided motivated with our findings to only analyse the combination of the top 3 materials and business 'Referencias Faro' logic, in the subactivities we found to be more important, ending with a dataframe of 320k rows, reducing it from the 2.5M we initially had.

```
The top 3 materials suppose: 0.1797749709039136
For subactivity 'RA' in the year 2022, the top 3 materials are:
A9K17616: 0.0850847253351223 y es REFERENCIA FARO
A9K17216: 0.04966627533065715
A9K17610: 0.045023970238134134
```

FIGURE 3.1: Top 3 Materials after computing weights for our Subactivity RA in the 2022. Own Source.

3.3 Feature Engineering

After cleansing the data, we initiated the process of Feature Engineering - a pivotal step to enhance the potency of our predictive models. This process consisted of transforming the raw data into features that better represented the underlying problem to the predictive models, resulting in improved model accuracy on unseen data. We applied several transformations, each tailored to cater to specific characteristics of our data:

1. **Data Type Conversion:** Certain columns in the dataset were converted into string format for consistency and ease of processing. These included 'Material', 'Subactividad', 'Cust_class', and 'tx_type'. The 'Date' column was converted into a datetime object to facilitate time-based operations.

2. **Time Feature Addition:** We enriched our data with time-related attributes such as the month and year of the transaction. Additionally, we added a 'time_idx' attribute to capture the temporal relationship between the records and a necessary feature for our Temporal Fusion Transformer model to have time consistency.

3. **Aggregated Feature Addition:** We added aggregated features to capture trends in the data. These included calculating the average volume by 'Cust_class', 'tx_type', and 'Subactividad' for each time period, which gives our models information based on these 3 categories as we have seen in the EDA Dashboard, this might has an effect on our results.

4. **Data Grouping:** The data was grouped by several key features including 'Material', 'Subactividad', 'Cust_class', 'tx_type', among others. Aggregated values for 'Price', 'qty', and 'ped_brutos' were calculated for each group. We do not want a granularity for each sales, instead we want data to be grouped in those categories to easily investigate forecasting and optimization.

5. **Price Feature Addition:** Several price-related features were added to better capture price trends and anomalies in the data. These included 'Precio_medio', logarithmic transformations of 'Precio_medio', 'Price', and 'Quantity', and 'Discount_pct', refering to the percentage of discount in every transaction. All logarithmic transformations were safeguarded with a negligible constant to avoid taking the logarithm of zero. These logarithms are going to be used in both TFT model and Price Optimization, following OLS standards [5].

6. **Minimum Encoder Length Assurance:** Time series in the dataframe were validated to have a minimum length, dictated by the encoder length and prediction length. This validation was crucial to ensure the integrity of our analysis, as only those series which had adequate past information for encoding and a satisfactory future horizon for predictions were retained.

7. **Filtering Based on Business Logic and Data Insights:** Our data was further refined based on the business knowledge and data insights we gained from our exploratory analysis. This included focusing on specific transaction types, 'top_ref' materials, and a select set of subactivities which had proven to have significant sales.

This comprehensive feature engineering, designed to capture the intricacies and patterns in the data, prepared us to embark on the next phase: training our models. Take note that this step has further reduced and filtered our dataset, now remaining only 11539 rows of information, accounting for the Top References in every month from the last 5 years and in the different transaction types and customer classification possibilities.

Chapter 4

Experiments and Results

This chapter is a cornerstone in our study, focusing on the analytical examinations and subsequent findings related to two major scopes: Price Elasticity and Sales Forecasting. The goal of this chapter is to present a comprehensive account of our conducted experiments, interpreting the results, and deriving meaningful insights that could influence strategic decision-making.

Within these scopes, we explore a variety of aspects, each with a unique significance and bearing on our overall analysis. To maintain clarity and ensure an organized presentation, this chapter has been divided into two main sections, each devoted to a distinct scope of our study.

4.1 Price Elasticity scope

In the Price Elasticity section, we will be discussing the nature of our experiment, the specific metrics we used to quantify and analyze elasticity, our initial baseline setup, and finally, the results obtained from the experiment.

4.1.1 Experiment 1: Ordinary Least Squares

Our initial investigation into price elasticity of demand involved the application of Ordinary Least Squares (OLS) regression on binned, log-transformed variables, a straightforward and transparent method that offers a preliminary understanding of the dataset at hand.

Our dataset, covering all material types, was subjected to log transformation for both price and quantity variables, following standard practice for elasticity studies [9]. This transformation was carried out as the elasticity of demand can often be approximated as a constant rate of change in quantity demanded relative to price, which is more naturally expressed in logarithmic terms. Additionally, log transformation has the added benefit of mitigating the potential effects of extreme values and ensuring a more linear relationship between price and quantity.

To handle the inherent variability and noise present in the dataset, we divided the range of our log-transformed price into 15 equal bins, averaging the log-transformed quantities within each bin. This 'binned OLS' approach served to reduce the influence of noise and outliers, providing a clearer visual depiction of the underlying trend between price and quantity.

The OLS regression on these binned averages revealed a negative relationship between log-transformed price and quantity, a manifestation of the law of demand that higher prices lead to lower quantities demanded. The estimated coefficient for log-transformed price stood at approximately $\theta = -0.588$. Interpreted in terms of price elasticity, this suggests that a 1% increase in price would lead to a roughly 0.588% decrease in quantity demanded, assuming all else constant. The R-squared value, a measure of goodness of fit, was found to be 0.789, indicating that the model was able to explain approximately 78.9% of the variability in the log-transformed quantity. The results can be seen in Figure A.11, the blue line is from the log-log experiment.

While this preliminary analysis serves as a useful starting point, it's important to remember that it falls into the category of a 'naive' analysis, in the sense that it does not control for potential confounding factors that could affect both price and quantity. The subsequent phases of our study will delve deeper, aiming to elucidate a more accurate picture of the causal relationship between price and quantity.

4.1.2 Experiment 2: Double Machine Learning

The **Double Machine Learning** (DML) approach provides a powerful technique for estimating causal effects in high-dimensional settings. It is particularly effective when the number of features is large or even exceeds the number of observations. DML excels at addressing the challenge of parameter estimation where high-dimensional nuisance parameters, although not of immediate interest, significantly affect the analysis.

The price elasticity of demand, computed using DML, is given by the coefficient β_1 from the OLS regression model, defined as the partial derivative of the natural logarithm of quantity (Q) with respect to the natural logarithm of price (P), i.e.,

$$\beta_1 = \theta = \frac{\partial \ln(Q)}{\partial \ln(P)}$$

Our results have shown a negative Elasticity in the figure A.12. The graph displays the "naive" approach, depicted in blue, which presents a basic correlation between log-quantity and log-price without any control for confounding variables. Despite the evident scatter, a negative relationship can be observed, signifying that an increase in prices corresponds to a reduction in demand, with the estimated regression coefficient being around $\theta \approx -0.6$.

The orange line offers a visual representation of the outcome derived from the Double Machine Learning (DML) technique, having adjusted for potential confounding influences. A steeper slope can be seen, suggesting an increased elasticity of demand ($\theta \approx -7$). Alongside this, there is a notable increase in the dispersion of the relationship, which is less than ideal. However, it is important to acknowledge that a significant decrease in price variance is evident (reflected in the reduced range of the line on the x-axis), attributable to the DML residualization process which accounts for much of this variation.

4.1.3 Experiment 3: Graph Causal Model

As a first step, we collected a comprehensive set of variables that we considered could influence both price and quantity. These variables include elements such as production cost, market competition, consumer income, and preferences, to name a few. This exhaustive list served as our nodes in the graph.

Next, we established the edges or causal relationships between these nodes based on theoretical considerations, prior knowledge, and initial data exploration. These directed edges represent our assumptions about the cause-and-effect relationships between variables. Two primary elements, 'Material' and 'Subactividad', directly influence both 'Price' (the original global cost) and 'Precio_medio' (the final price), with 'Price' further impacting 'Precio_medio'. The final price is also influenced by 'Cust_class', the customer classification, and 'tx_type', the type of transaction.

'qty', representing quantity or volume, is shaped by temporal variables ('time_idx' and 'month'), transaction type, customer class, and the final price, signifying its sensitivity to various factors. See Figure A.13 where all the variables involved are represented.

Given the complexity of real-world economic systems, we made simplifying assumptions about the causal graph's structure to aid in estimation. We assumed a "causal sufficiency" condition, that is, there are no common causes of any pair of variables in our graph that we have not included.

We took our constructed Generative Causal Model (GCM) and applied it to the dataset we gathered, producing an effective 'causal version' of the data. This allowed us to simulate the causal effects of different variables on our target outcome: demand. A central part of this process was computing the Price Elasticity of Demand (PED).

To calculate PED, we simulated a series of price changes in our model and observed the resulting changes in quantity demanded. The computed PED values provide an estimate of the percentage change in demand that would result from a 1% change in price. For increased precision, we considered a range of percentage changes, not merely a 1% alteration. With the computed PED, we could estimate how changes in price might affect demand, giving us a tool for strategic decision-making.

Finally, we conducted a backtesting process on our data to find the price that maximizes revenue. We considered a range of percentage changes in price and simulated their effects on demand and, consequently, revenue. Through this process, we found the optimal price point that maximizes revenue for each data point in our dataset. In addition, we computed the corresponding quantity and revenue at these optimal price points. This rigorous backtesting procedure helps to validate our model's effectiveness in maximizing revenue while also providing practical, data-driven insights for pricing strategies. We will present the final results in the section 4.3.2. It's important to emphasize that our objective is neither prediction nor classification but rather adjusting historical data. Therefore, gauging the precision or accuracy of our model can be particularly challenging.

4.2 Sales Forecasting scope

Subsequently, we delve into Sales Forecasting, a fundamental part of business strategy that allows for effective planning and resource allocation. Similar to the previous section, we will introduce the experiments.

4.2.1 Experiment 1

Firstly, the data is filtered based on certain conditions. In this basic experiment we tried to find patterns only for a concrete transaction type because of business beliefs of being the most stable and logical one. Not only this but we did not provide any further information more than temporal one, price and quantity. Before training the TFT model, the mean absolute error (MAE) of a simple naive baseline model, predicting the next value as the last available value from the history, is calculated for future comparison with the TFT model.

The TFT model is then configured and trained. It's worth noting that to improve training, several techniques are used, including early stopping to prevent overfitting, learning rate monitoring for better optimization, and logging with TensorBoard for tracking the training process. The model is trained with a particular set of hyperparameters including hidden size, dropout rate, learning rate, and others. Obviously for a time series, we usually do not shuffle the training set, and the division has been done leaving 6 months for test, and the rest for train and validation.

After the model is trained, its performance is evaluated by calculating the mean absolute error on the validation set. The results show an improvement over the simple baseline model but the error is yet too high, which makes us change the data fed.

4.2.2 Experiment 2

In this second experiment we changed and enriched the dataset because of the complexity inherited in the data in the first experiment, the model was not able to see the different patterns for different customer classifications. Therefore in this experiment, we trained the Transformer over all the data, using the top references as previously stated, and we used as group_ids[1] the Material, Customer Classification and Transaction Type.

We will see in our results 4.3.2 that this is always the minimum classification because we discovered it was impossible to see any kind of patterns in a higher granularity. We also used more features, like the percentage of discount, the monthly average volume by Customer type, by transaction type and by subactivity. We also incorporated a lag in quantity of 1,3 and 6 months.

4.2.3 Results

In Experiment 1, our initial benchmark is set by the Baseline model, a rudimentary forecasting approach predicting the subsequent value as the final observable value from historical data. This simplistic model produces a mean absolute error (MAE) of 1193.29. In contrast, the Temporal Fusion Transformer (TFT) model demonstrates an improved MAE during the training stage (405.0), which however increases during the validation (943.0) and testing phases (1479.28). This trend might suggest an overfitting scenario, where the model is excessively tailored to the training data, diminishing its generalization capabilities on unseen validation and test data.

Upon integrating hyperparameter tuning into the TFT model, we observe a substantial improvement across all stages — with MAE results for training, validation, and testing being 364.0, 693.0, and 1111.55 respectively. This enhancement is attributed to the optimization of the model's configurations specific to the task at hand via hyperparameter tuning, which subsequently boosts the model's performance.

Experiment 2 follows a similar pattern albeit with significantly lower MAE values. The baseline model here yields an MAE of 367.51. The TFT model, akin to Experiment 1, exhibits an escalating MAE from training (102.0) to validation (254.0) and testing phases (408.49), potentially suggesting an overfitting issue. However, the lower MAEs relative to Experiment 1 implies that the data is less complex due to the reason we are grouping by Material, transaction type and customer classification, making it more susceptible to effective modeling with the TFT approach.

When incorporating hyperparameter tuning in Experiment 2's TFT model, we again notice enhanced performance, albeit the MAE shows a mild increase from training (203.0) to validation (231.0) and testing stages (367.61). Interestingly, the testing MAE of the hyperparameter-tuned TFT model approximates that of the baseline

[1]Group Ids allow the TFT model to divide the time series in groups, dealing with each group independently from the others.

TABLE 4.1: Experiment Results. Metric used is MAE

	Baseline	TFT			TFT w/ Hyperparameter Tuning		
		Train	Validation	Test	Train	Validation	Test
Experiment 1	1193.29	405.0	943.0	1479.28	364.0	693.0	1111.55
Experiment 2	367.51	102.0	254.0	408.49	203.0	231.0	367.61

model in Experiment 2. This observation may suggest an inherent limitation in the performance achievable with the given data and task setup, at least when using this specific model with its current hyperparameter configuration. All the exact results can be seen in the Table 4.1

To finish the results, the Temporal Fusion Transformer allows us to see the importance for each part. In the Figure A.16, we can see how the temporal attention follows a logical structure, giving more importance to more recent data. The encoder and decoder importances make also sense, while the encoder gives more importance to the lagged features of quantity, the decoder gives more importance to the month for a possible seasonality we believe and features like the final price, indicating a strong relationship between quantity-price.

4.3 Final results and Presentation

In this last section we will see how our results are combined and how do we present the final results to the business, giving them a tool to interactively see which decisions should they take.

4.3.1 Combining both models

Our aim is to elucidate the combination of different models used to optimize Schneider Electric's (SE) time series data, which concludes in April 2023. Initially, Graph Causal Models (GCMs) were employed to optimize SE's time series data, enabling the identification of optimal price points for each month to maximize revenue. This optimization was performed within our predetermined range of prices, ensuring realistic transitions in pricing; substantial increments such as 100% month-to-month increases were strictly avoided. This results in what we define as the 'Optimized' time series, which concludes in the same month as the original SE time series and provides an optimized price for that month aimed at revenue maximization.

The next step in our method involves feeding this optimized price back into the original SE time series. We utilize the Temporal Fusion Transformer (TFT) model to forecast the optimal price for the final month that would result in maximizing both revenue and the quantity sold for the subsequent six months.

By integrating these methodologies, we can simultaneously optimize historical data and generate future predictions. This approach enriches our understanding of price elasticity for key materials within a specific customer and transaction type, offering valuable insights into how commercial negotiations could be more effectively conducted. Moreover, it provides an estimation of future sales under different pricing strategies: a) maintaining current pricing as per the original SE time series, b) shifting to optimized pricing based on historical data, and c) adopting optimal pricing for maximizing either future revenue or volume. This comprehensive approach thus holds significant implications for price negotiations and sales forecasting.

4.3.2 Presentation of results

In conclusion, we have emphasized an essential, yet often overlooked component in the realm of Data Science - the effective presentation of results. Recognizing the indispensability of this aspect from the onset of our project, we employed the Tableau software to create a series of dashboards. Some of these, as introduced in Chapter 3.2, offer rich insights derived from the comprehensive data compilation and cleaning process, a novel undertaking for Schneider Iberia. These interactive and comprehensible dashboards not only elevate the value of our work but also provide a beneficial tool to aid business decisions.

Finally, we present two concluding dashboards designed to elucidate the results derived from our models. The inaugural dashboard, referenced as the Optimization dashboard in Figure 4.1, comprises four graphs and the customary filters of Material, Customer, and Transaction type. The first graph elucidates the Price Elasticity of Demand (PED), indicating in this particular instance that the material exhibits inelastic behavior. This information implies that Schneider could potentially raise prices without adversely affecting sales, given the observed inelasticity. The subsequent graphs illustrate, in green, the original pricing, volume, and resultant revenue as per Schneider Electric's policy. Conversely, the red lines represent the prospective outcomes given optimal pricing. These illustrations reveal that the red revenue (representing optimized pricing strategy) consistently surpasses the original green revenue, thereby highlighting the potential benefits of the optimized pricing approach.

The second dashboard, in Figure 4.2, is engineered to guide business decisions concerning optimal pricing strategies. As discussed in the previous subsection, we now have at our disposal 3 distinct time series: the original Schneider time series, and two novel series depicting optimum outcomes for revenue and volume, respectively. This dashboard provides a comparative analysis of these time series, showcasing their predicted behavior in terms of price, volume, and resultant revenue over a future timeframe. The dash vertical line indicates the last known datapoint. Note that we deploy the Temporal Fusion Transformer (TFT) to forecast volume, hence we keep prices constant as their evolution cannot be accurately predicted.

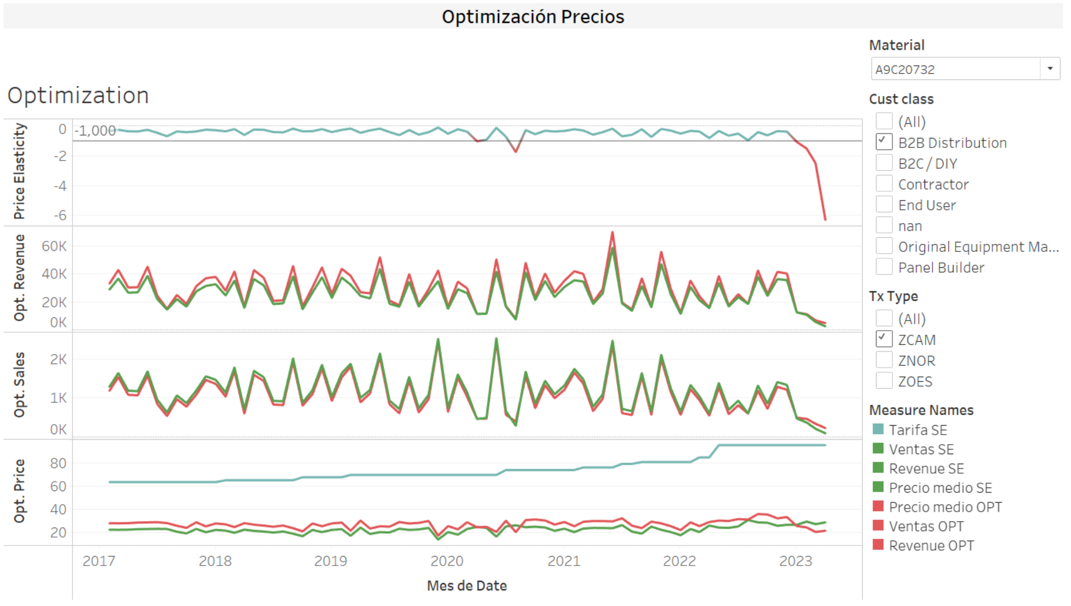


FIGURE 4.1: Dashboard for Price Optimization. Own Source.

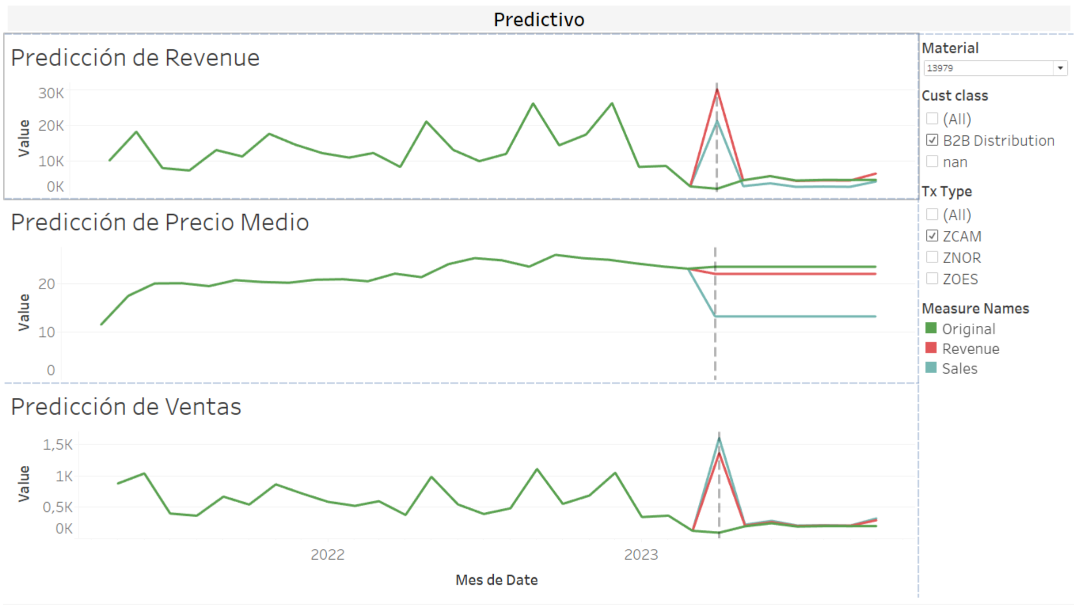


FIGURE 4.2: Dashboard for Sales Forecasting. Own Source.

Chapter 5

Conclusions and future work

In this thesis, the primary objective was to employ data science methodologies to address a pertinent real-world problem within the context of Schneider Electric, a leading multinational corporation. The successful fulfillment of this objective is attributed to the rigorous execution of various phases of data science, including meticulous data collection, preprocessing, and exploratory data analysis.

Data collection and preprocessing incorporated not only insightful data understanding but also the implementation of business logic. An integral part of this process was ongoing communication with key individuals in Schneider Electric's Iberia division, which allowed for the successful realization of the project. Equally important was the translation of complex data into intuitive and meaningful visualizations, demonstrated in Chapter 3.2.

Upon successful completion of the initial phase of data collection and cleaning, this study implemented a causality-based model to determine elasticity and optimize these materials. This innovative approach involved comparisons with other experiments such as ordinary least squares (OLS) and double machine learning. While it is challenging to ascertain the realism of the counterfactual world due to multiple confounding variables, the data and insights shared by the Pricing Heads at Schneider suggest that most confounding variables have been adequately addressed.

In parallel, a state-of-the-art model, Temporal Fusion Transformer (TFT), was utilized for volume prediction. However, the complexities and inherent randomness in the data may have limited the model's predictive power. This limitation can be attributed to the business model of Schneider's clients, most of whom operate on a business-to-business (B2B) model. The indirect nature of these relationships introduces a level of separation that could be impacting prediction accuracy. Future work should further explore these time series, potentially employing more exhaustive feature engineering, testing other models, and exploring the possibility of using foundational or pretrained models to discern latent patterns.

Ultimately, this thesis presents a complete end-to-end project that not only visualizes the results of both the causality-based and TFT models but also provides business insights into Schneider's data. These insights include recognizing patterns such as stable final prices for clients despite increasing list prices, and identifying client segments that buy more of certain transaction types or subactivities. Such information, often implicit and held by long-standing department members, has been explicitly delineated and visualized in this study. This knowledge is invaluable for Schneider Electric as it embarks on future business strategies and decisions.

As for future directions, the immediate task would be to enhance the performance of the sales forecasting model. Additionally, the applicability of this pipeline to other Helios activities could be explored. Given the similar or identical data structure, the data collection and feature engineering pipeline should not necessitate major alterations. However, any successor to this project must remain cognizant of the critical role of stakeholders and business developers in interpreting and assessing the insights derived logically. Consequently, each Helios activity should be examined in conjunction with the respective department head to determine suitable courses of action.

A potential improvement could be the implementation of continuous integration/-continuous deployment (CI/CD) procedures. This would enable the ingestion of the most recent month's data into the pipeline and facilitate monthly retraining of the model to forecast the upcoming two quarters. While this final step of model deployment was not realized due to constraints of time and complexity, it represents a worthwhile endeavor for future work.

List of Figures

Appendix A

Annex

Link to the Github repository of our MSc Thesis, please read the README file.

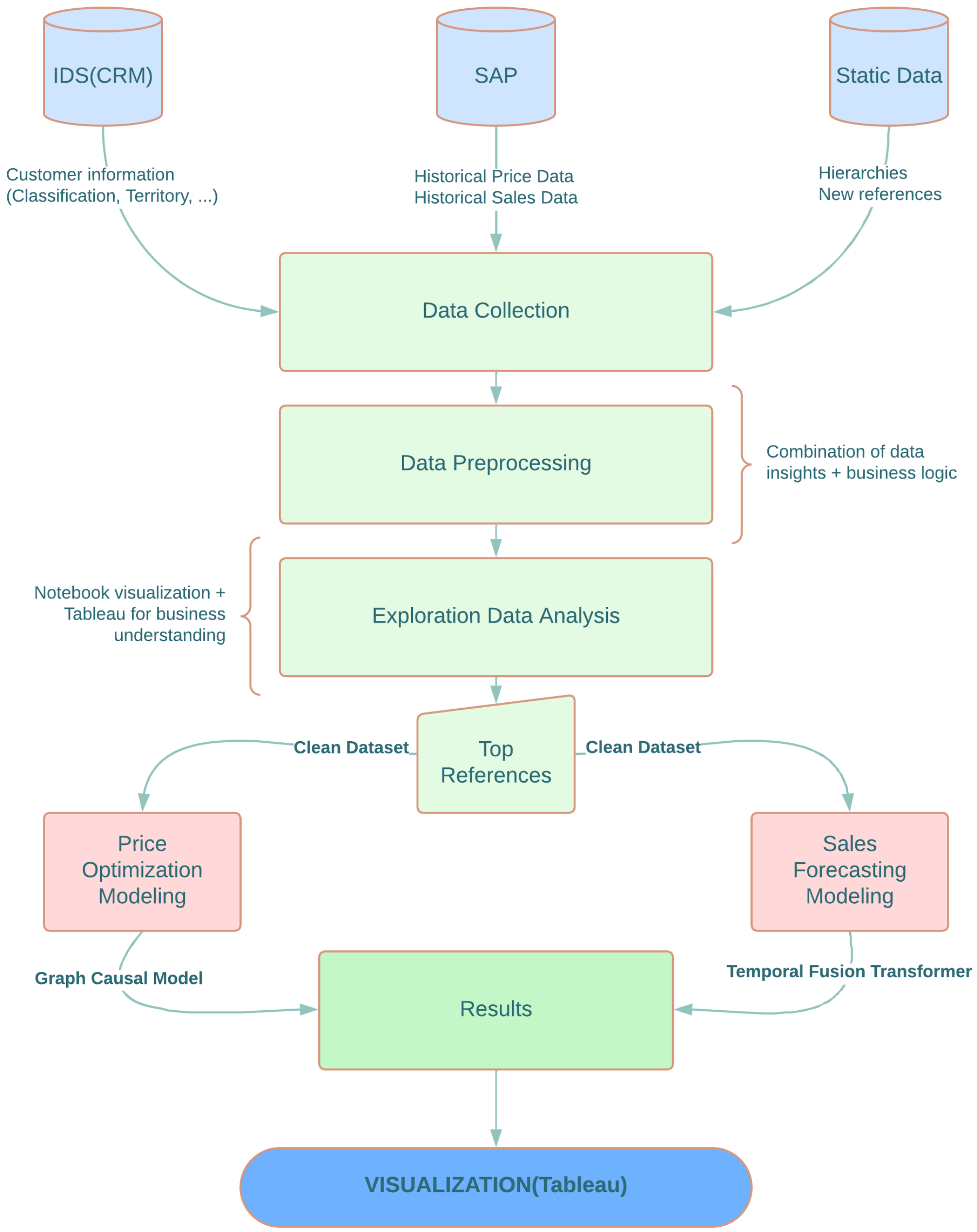


FIGURE A.1: Workflow of our end-to-end Data Science Project. Own Source

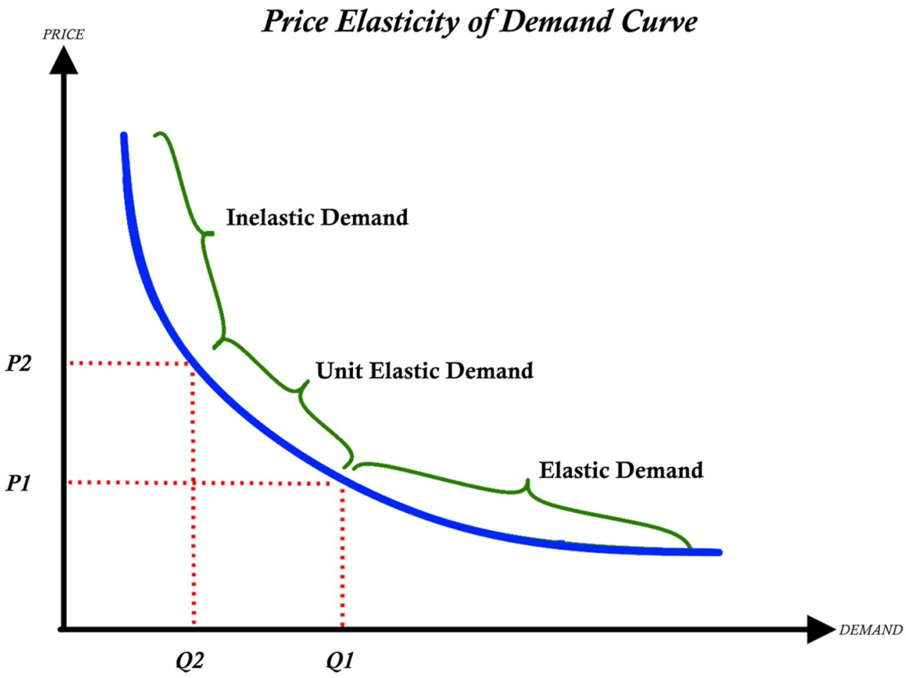


FIGURE A.2: Elasticity of Demand, visual representation. [20]

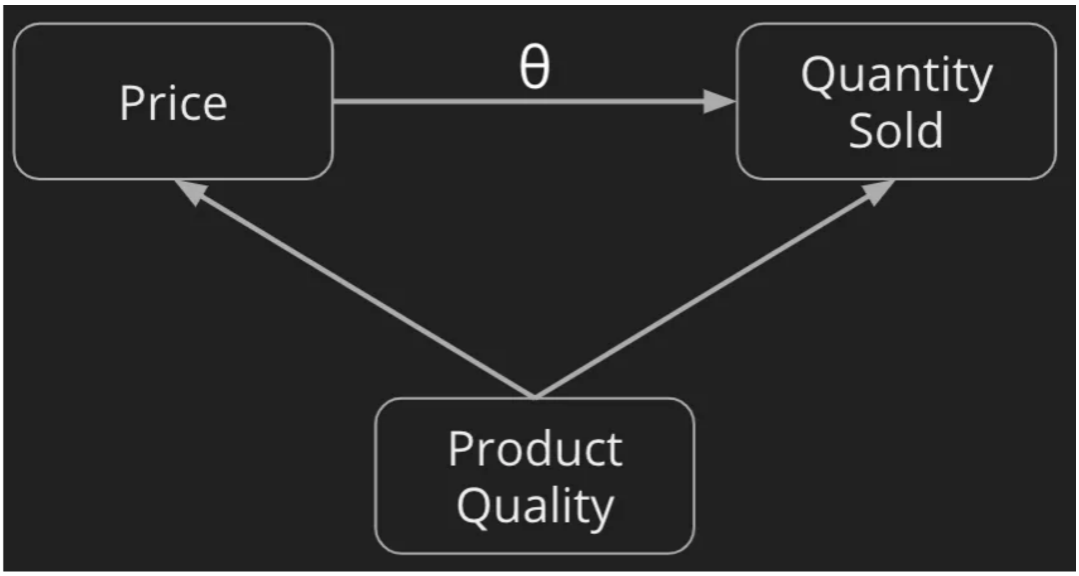


FIGURE A.3: A minimum (i.e. incomplete) causal graph between price and quantity, with product quality as only confounder. θ refers to the Elasticity [9]

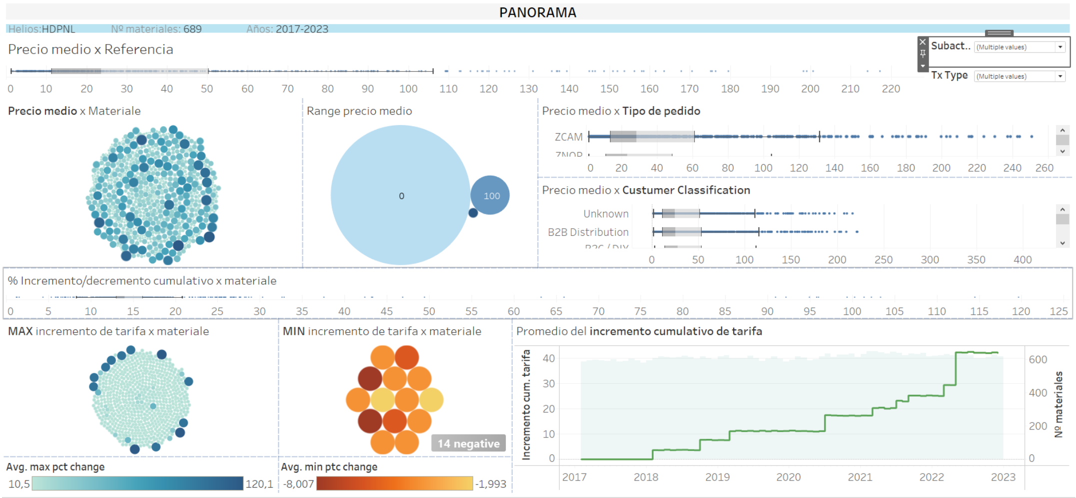


FIGURE A.4: General view of the EDA Dashboard. Own Source.

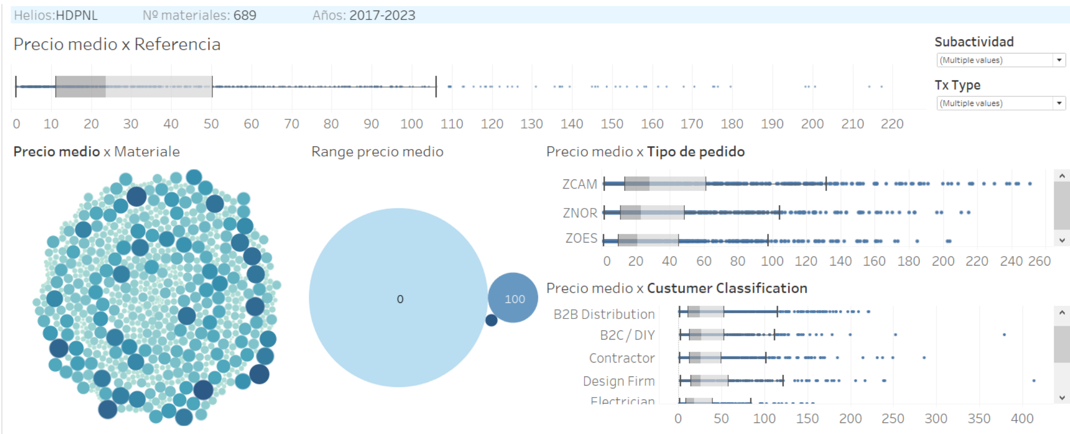


FIGURE A.5: First part of the Dashboard. Own Source.

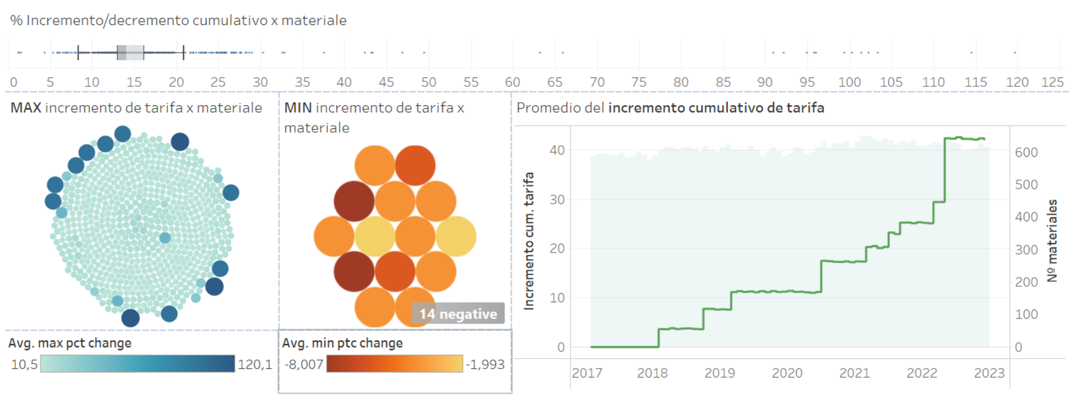


FIGURE A.6: Second part of the Dashboard. Own Source.

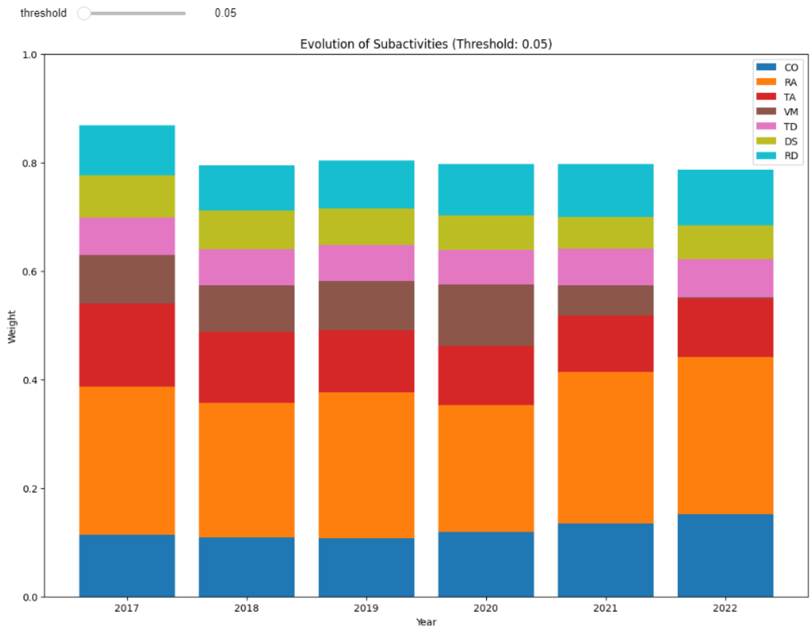


FIGURE A.7: Top Subactivities in HDPNL Helios, only showing the ones with higher weight than 5%. Own Source.

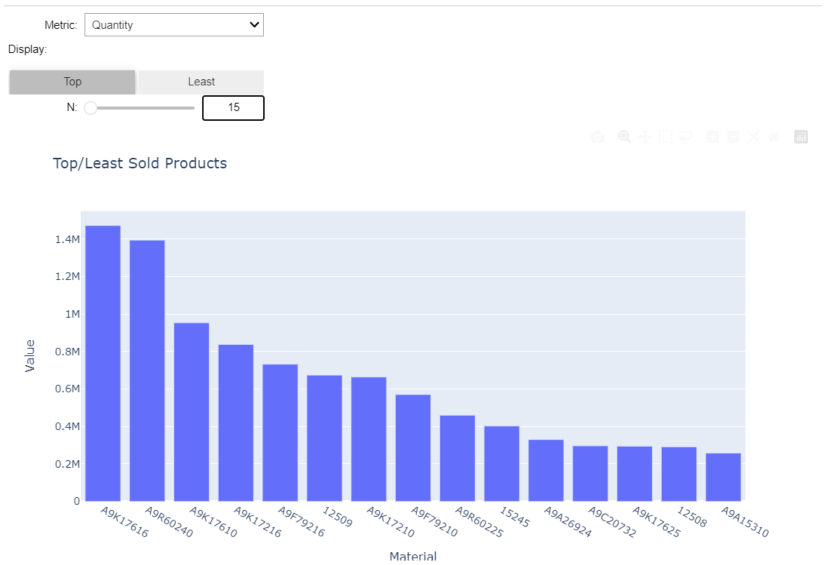


FIGURE A.8: Top Materials in the whole Helios, showing the top 15. Own Source.

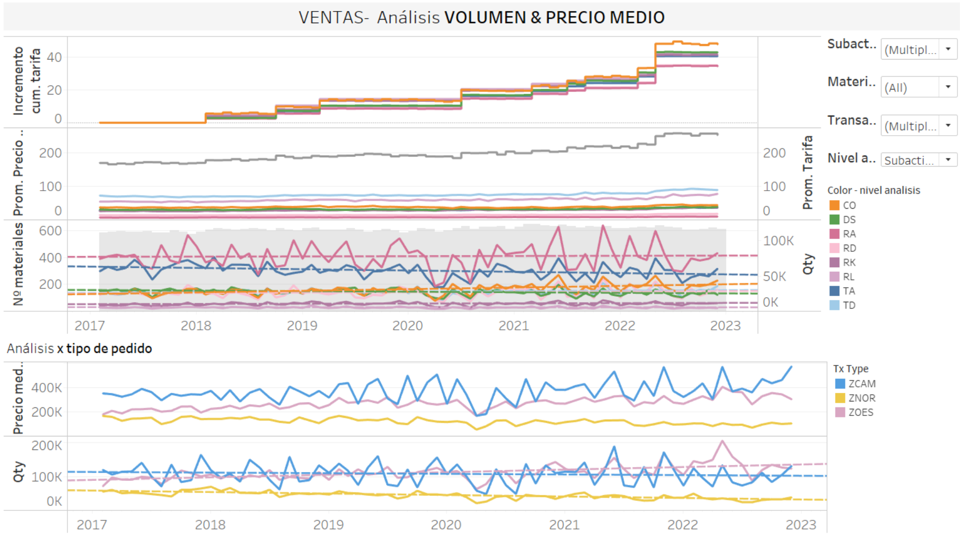


FIGURE A.9: Dashboard analyzing sellings and prices. Own Source.

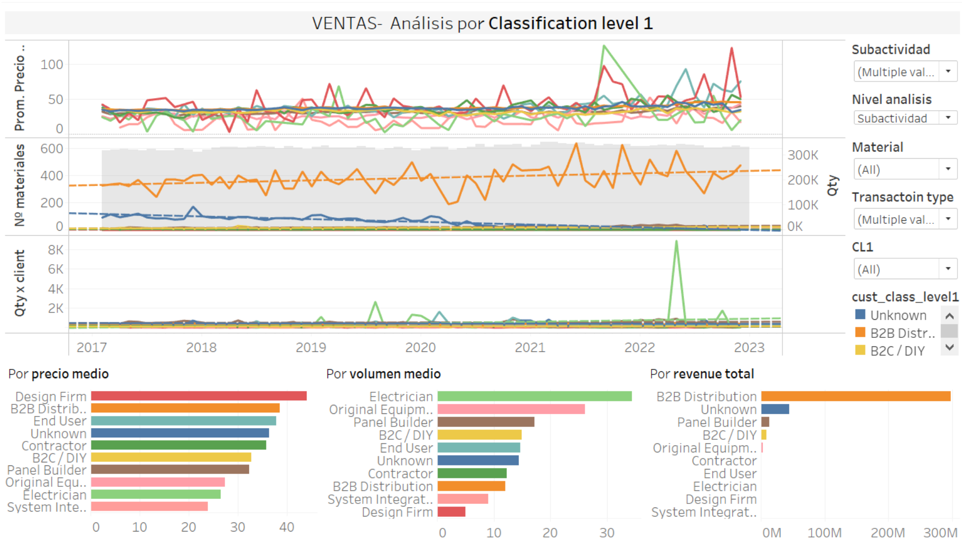


FIGURE A.10: Dashboard analyzing customer behavior. Own Source.

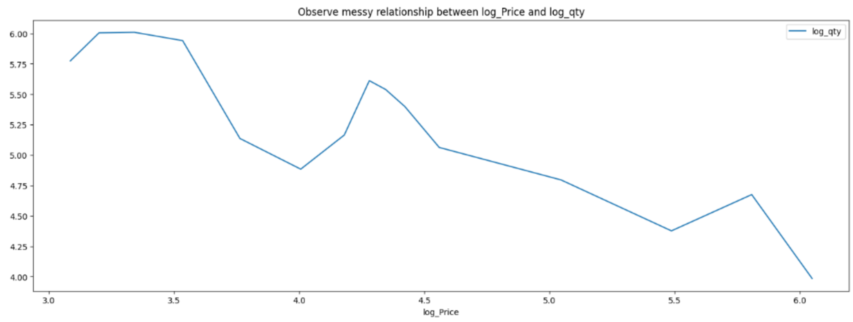


FIGURE A.11: OLS results for Elasticity. Own Source.

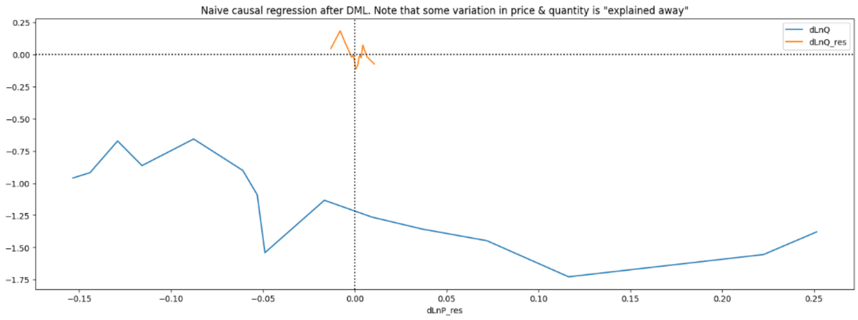


FIGURE A.12: DML results for Elasticity. Own Source.

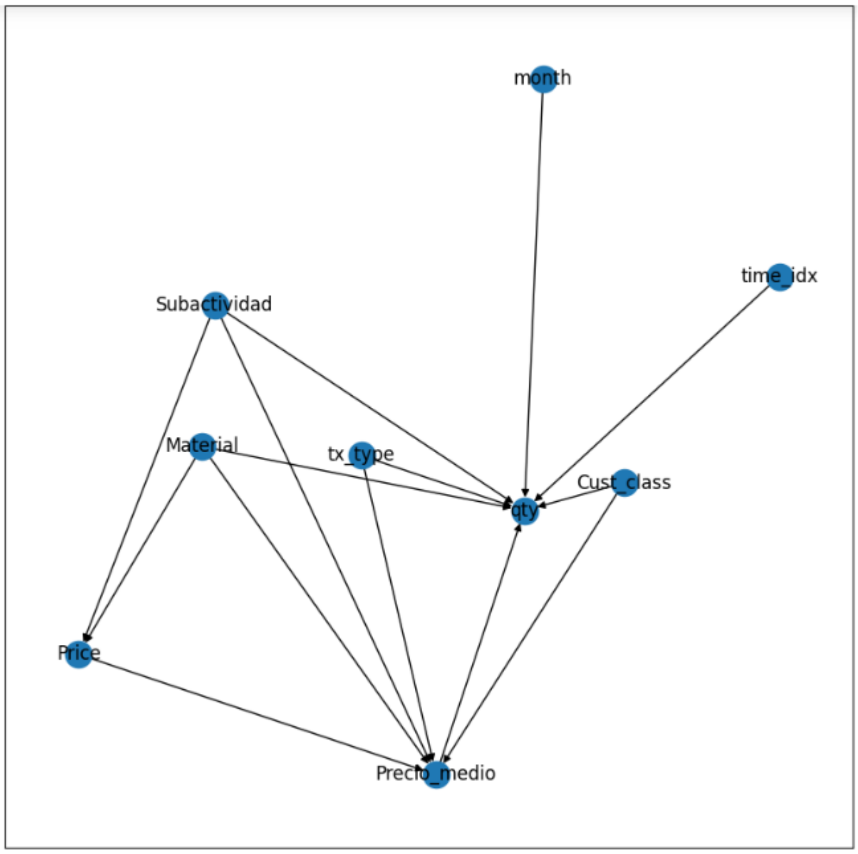
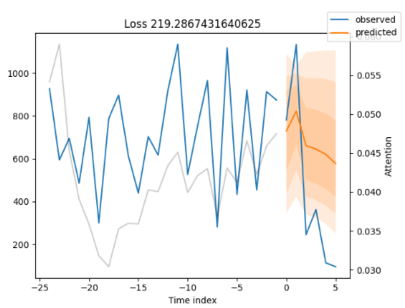
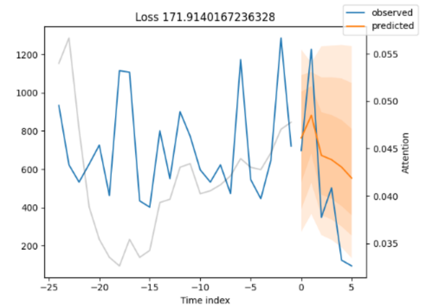


FIGURE A.13: GCM structure for Elasticity. Own Source.

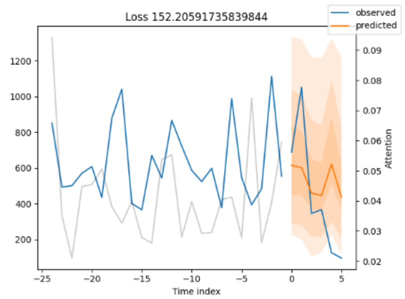


(A) Results on validation data for experiment 1 using the TFT.

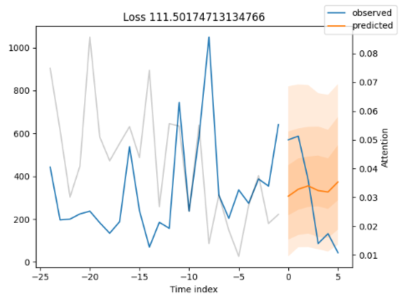


(B) Results on validation data for experiment 1 using the TFT.

FIGURE A.14: Results for experiment 1.

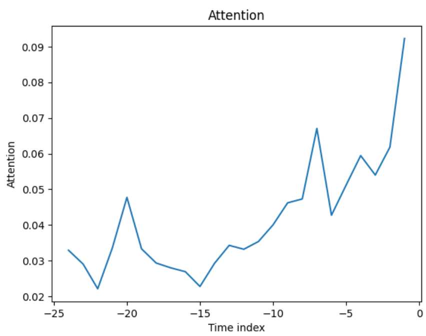


(A) Results on validation data for experiment 2 using the TFT.

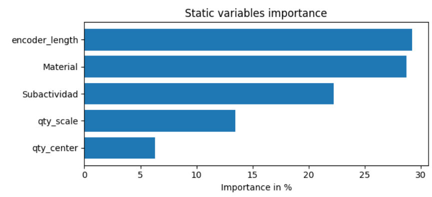


(B) Results on validation data for experiment 2 using the TFT.

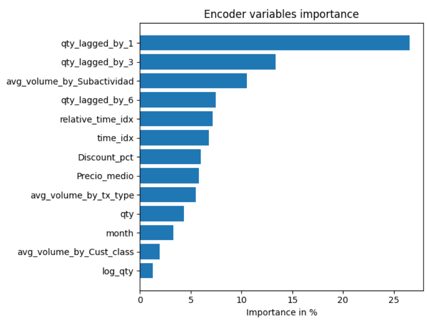
FIGURE A.15: Results for experiment 2.



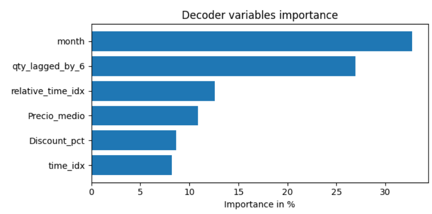
(A) Temporal Attention.



(B) Static Variable importance.



(C) Encoder importance.



(D) Decoder importance.

FIGURE A.16: Importances in Experiment 2

Bibliography

[1] A. Marshall. *Principles of economics*. English. London: Macmillan, 1890.

[2] R.H. Frank. *Principles of economics*. English. New York, NY: McGraw-Hill Education, 2015.

[3] Jeffrey M. Wooldridge. *Introductory Econometrics: A Modern Approach*. 7th. South-Western College Pub, 2019. ISBN: 978-1337558860.

[4] Victor Chernozhukov et al. "Double/debiased machine learning for treatment and structural parameters". In: *The Econometrics Journal* 21.1 (2018), pp. C1–C68. DOI: 10.1111/ectj.12097.

[5] Fabian Dablander. "An Introduction to Causal Inference". English. In: *Unspecified* (Unspecified).

[6] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*. 2nd ed. Springer, 2009. ISBN: 978-0-387-84857-0.

[7] Kevin D. Hoover. *Causality in Economics and Econometrics*. An Entry for the New Palgrave Dictionary of Economics. Accessed: 2023-06-14. Unspecified.

[8] Matheus Facure. *Python Causality Handbook: Graphical Causal Models*. https://matheusfacure.github.io/python-causality-handbook/04-Graphical-Causal-Models.html. Accessed: 2023-06-14. Publisher/Website, 2023.

[9] Lars Roemheld. *Causal Inference in the Wild: Elasticity Pricing*. https://towardsdatascience.com/causal-inference-example-elasticity-de4a3e2e621b. Accessed: 2023-06-14. Towards Data Science, 2021.

[10] J.T. Mentzer and M.A. Moon. "Understanding demand: forecasting". English. In: *Business forecasting: practical problems and solutions*. 2001, pp. 455–471.

[11] R.J. Hyndman and G. Athanasopoulos. *Forecasting: principles and practice*. English. OTexts, 2018.

[12] A. Vaswani, N. Shazeer, N. Parmar, et al. "Attention is all you need". English. In: *Advances in neural information processing systems*. 2017, pp. 5998–6008.

[13] Bryan Lim et al. *Temporal Fusion Transformers for Interpretable Multi-horizon Time Series Forecasting*. https://arxiv.org/pdf/1912.09363.pdf. Accessed: 2023-06-22. 2020.

[14] Sana Prasanth Shakti et al. "Annual Automobile Sales Prediction Using ARIMA Model". In: *International Journal of Hybrid Information Technology* 10.6 (2017). Accessed: 2023-06-23, pp. 13–22. DOI: \url{https://doi.org/10.14257/ijhit.2017.10.6.02}.

[15] Kyoung-jae Kim. "Financial time series forecasting using support vector machines". In: *Neurocomputing* 55.1-2 (2003). Accessed: 2023-06-23, pp. 307–319. DOI: \url{https://doi.org/10.1016/S0925-2312(03)00372-2}.

[16] Alaa Sagheer and Mostafa Kotb. "Time series forecasting of petroleum production using deep LSTM recurrent networks". In: *Neurocomputing* 323 (2019). Accessed: 2023-06-23, pp. 203–213. DOI: \url{https://doi.org/10.1016/j.neucom.2018.09.082}.

[17] Chaitanya K. Joshi et al. "Learning the travelling salesperson problem requires rethinking generalization". In: *Constraints* 27 (2022). Accessed: 2023-06-23, pp. 70–98. URL: \url{https://link.springer.com/article/10.1007/s10601-022-09369-3}.

[18] Yun Peng and James A. Reggia. "A Probabilistic Causal Model for Diagnostic Problem Solving Part I: Integrating Symbolic Causal Inference with Numeric Probabilistic Inference". In: *IEEE Transactions on Systems, Man, and Cybernetics* 17 (2 Unspecified). Accessed: 2023-06-23.

[19] Tanmayee Narendra et al. "Counterfactual Reasoning for Process Optimization Using Structural Causal Models". In: *Business Process Management Forum*. Vol. 360. Lecture Notes in Business Information Processing. Accessed: 2023-06-23. International Conference on Business Process Management. 2019, pp. 91–106.

[20] Happy Happy. *Elasticity : Elasticity of Demand | Definition | Economics | Formula*. https://www.excel-pmt.com/2019/01/elasticity-elasticity-of-demand.html. Accessed: 2023-06-14. Excel PMT - Financial Management, 2019.

[21] Guido W. Imbens and Donald B. Rubin. *Causal inference in statistics, social, and biomedical sciences*. English. Cambridge University Press, 2015.

[22] Susan Athey. "The state of applied econometrics: Causality and policy evaluation". English. In: *Journal of Economic Perspectives* 31.2 (2017), pp. 3–32.