September 1, 2025

Gokul Chirayath sudheer

U2734832@UEL.AC.UK - MSc Data SCience

Dissertation Guide: Patrick OXENHAM

Price Elasticity Analysis and Pragmatic Price Optimization for E-commerce SME

DS7010 - Dissertation

Table of Contents

[Abstract 3](#_Toc207659725)

[1. Introduction 4](#_Toc207659726)

[1.1 Background of the Study 4](#_Toc207659727)

[1.2 Research Aim and Objectives 5](#_Toc207659728)

[1.3 Research Questions 5](#_Toc207659729)

[2. Literature Review 6](#_Toc207659730)

[2.1 Introduction 6](#_Toc207659731)

[2.2 The Econometric Foundations: Estimating Price Elasticity 6](#_Toc207659732)

[2.3 The Paradigm Shift to Machine Learning in Pricing Analytics 8](#_Toc207659733)

[2.4 From Prediction to Action: Optimization and Business Constraints 9](#_Toc207659734)

[2.5 Bridging the Interpretability Gap in Machine Learning 9](#_Toc207659735)

[2.6 Synthesis and Research Gap 10](#_Toc207659736)

[3. Methodology 12](#_Toc207659737)

[3.1 Research Philosophy and Design 12](#_Toc207659738)

[3.2 Data and Sampling Methodology 12](#_Toc207659739)

[3.3 Data Preprocessing and Feature Engineering 13](#_Toc207659740)

[3.3.1 Initial Cleaning and Filtering 13](#_Toc207659741)

[3.3.2 Temporal Aggregation 14](#_Toc207659742)

[3.3.3 Outlier Handling 14](#_Toc207659743)

[3.3.4 Feature Engineering 14](#_Toc207659744)

[3.4 Statistical/machine learning models 15](#_Toc207659745)

[3.4.1 Explanatory Model: Log-Log Regression 15](#_Toc207659746)

[3.4.2 Model for Prediction: Random Forest Regressor 15](#_Toc207659747)

[3.4.3 Predictive Benchmark: XGBoost Regressor 15](#_Toc207659748)

[3.5 Model Optimisation and Tools 16](#_Toc207659749)

[3.6 Evaluation Strategy 16](#_Toc207659750)

[3.7. Ethical Considerations and Reliability 17](#_Toc207659751)

[4. Implementation 17](#_Toc207659752)

[4.1 The Analytical Workflow: From Raw Data to Actionable Insight 17](#_Toc207659753)

[4.2 Data Preprocessing and Exploratory Analysis 18](#_Toc207659754)

[4.3 Feature Enhancement and Dataset Finalisation 20](#_Toc207659755)

[4.4 Comparative modelling and systematic optimisation 21](#_Toc207659756)

[4.4.1 Baseline Model: The Explanatory Log-Log Regression 21](#_Toc207659757)

[4.4.2 Predictive Modelling: An Initial Challenge and Critical Finding 22](#_Toc207659758)

[4.4.3 The Solution: Systematic Optimisation 23](#_Toc207659759)

[4.4.4 Final Model Training and Preparation for Evaluation 24](#_Toc207659760)

[5. Results, Discussion, and Analysis 26](#_Toc207659761)

[5.1 Presentation of Core Results 26](#_Toc207659762)

[5.1.1 Explanatory Model Results: Log-Log Regression 26](#_Toc207659763)

[5.1.2 Predictive Model Results: Tuned Machine Learning Models 27](#_Toc207659764)

[5.2 Full Comparative Performance Analysis 30](#_Toc207659765)

[5.3 Results Discussion 31](#_Toc207659766)

[5.4 Critical Analysis 31](#_Toc207659767)

[5.4.1 Strengths and Weaknesses of the Hybrid Approach 32](#_Toc207659768)

[5.4.2 Reliability, Validity, and Surprising Findings 32](#_Toc207659769)

[6. Conclusion and Future Work 34](#_Toc207659770)

[6.1 Conclusion 34](#_Toc207659771)

[6.2 Limitations 35](#_Toc207659772)

[6.3 Future Work 36](#_Toc207659773)

[References 37](#_Toc207659774)

[APPENDIX 38](#_Toc207659775)

# Abstract

Small and Medium-sized Enterprises (SMEs) in the E-commerce industry have difficulties to develop successful data-driven pricing strategies as they are caught between explainable and simple models which lack high predictive capabilities and accurate complex models which lack transparency. This deficit is addressed by this thesis, which develops, applies and evaluates a pragmatic hybrid analytical approach that combines the benefits of both the econometric and machine learning approaches.

Based on the Online Retail II transactional dataset, this study adopts quantitative quasi-experimentation. All the methods are implemented within a comprehensive data science pipeline that includes careful data preparation, extensive feature engineering, and outlier treatment. An Explanatory Log-Log regression model to estimate Price Elasticity of Demand (PED) is compared with two extremely accurate predictive algorithms, via Random Forest Regressor and an XGBoost Regressor, which are optimized through systematic hyperparameter tuning.

The primary results indicate a distinct split of usefulness. Although the Log-Log model does not perform well as a predictor, it offers the decision maker a statistically significant and interpretable measure of price sensitivity to use strategically. In contrast, the Tuned Random Forest model always performs better and can be viewed as the better tactical analysis tool, attaining a R-squared of 0.373 of the tests set for ’REGENCY CAKESTAND 3 TIER’. The study recommends a hybrid form of model where the Log-Log model is used for strategic “Why” insights and Random Forest is used for tactical “What” If predictions provide the best and most easy to use price reference system for the SME sector. This study offers a strong proof-of-concept, showing that a powerful dual-model approach is feasible for storefronts (low tech readiness), enabling a smooth transition between interpretability and prediction performance.

# 1. Introduction

The impact of online shopping has successfully transformed the retail landscape into an ultra-competitive environment with price transparency and low consumer switching costs. As digital has come to define the marketplace, pricing has changed from being a mundane operational task to a strategic lever which often decides the difference between success and failure. The large companies invest highly in sophisticated algorithmic pricing engines and full-time data science teams, but many Small and Medium-sized Enterprises (SMEs) have been left behind by a growing divide on analytics. These smaller enterprises typically can't move beyond simple cost-plus or follow-the-leader pricing strategies that are simply unequipped to deal with the vagaries of online consumer behaviour while also leaving money on the table and market share on the shelf.

This gap is directly targeted in this thesis by creating and validating a practical, data-driven hybrid method for price optimisation that is specifically tailored to the operational world of SMEs. At the heart of this work is a comparative study that exploits a large real-world transactional data source to study the trade-offs between traditional econometric approaches and more modern machine-learning methods. “The idea is to deliver a solution that is not just effective but is something that makes sense to the untrained eye of for the business that lacks specialist data science skills”.

## 1.1 Background of the Study

This study is based on the economic **Price Elasticity of Demand (PED)** theory, which describes the responsiveness of quantity demanded to change in price. The log-log linear regression model has long been the established way to estimate the price elasticity of demand (PED) from observational data. The remarkable success of this model is the elegant interpretability that the coefficient for the price variable gives: a direct, managerially meaningful estimate of elasticity. Initial meta-analyses of this latter kind were performed by Tellis (1988) and subsequently by Bijmolt et al. (2005) have established this alternative as a reference method in academia as well as in industrial routines.

But the recent surge in granular transactional data and compute power has brought about a transformation in pricing analytics to be more ML driven. In contrast to econometric models, which are aimed at inference and explanation, ML algorithms like Random Forest or XGBoost are tuned for prediction only. They can reveal intricate non-linear structures in data and offer improved predictive accuracy–a result that benefits businesses. But that predictive capability often comes at a price and many of these ML models are “black boxes” to business users, which means that we struggle to interpret them? This poses a critical dilemma for SMEs: a trade-off between a simple, comprehensible model likely to be inaccurate and a complex, trustworthy model that may not be up to implementation. This trade-off is the crux of this thesis, we intend to settle into a working combination of both paradigms which we find satisfactory.

## 1.2 Research Aim and Objectives

The main objective of this study is to construct, implement, and test a hybrid analytical methodology which facilitates the explanation capacity of econometrics and the accuracy prediction of machine learning to propose a practical price optimisation solution for e-commerce SMEs.

For this purpose, the following objectives have been formulated:

* The aim is to conduct a critical literature review of both econometric and (2) Machine learning and demand forecasting econometric PED estimation application of machine learning to demand forecasting.
* To build a solid data process pipeline to pre-process, aggregate and feature engineer a massive raw transactional dataset.
* To apply a log-log regression model to compute a statistically significant and interpretable measure of Price Elasticity of Demand.
* To develop, calibrate and validate (using statistical measures) two of the most popular machine learning algorithms (Random Forest and XGBoost) to obtain weekly sales predictions efficiently.
* To compare the three model types in a comprehensive manner and in this way provide an in-depth look at trade-offs amongst their explanatory and predictive performance.
* To combine these findings into a useful dual model structure that can offer actionable pricing advice to SMEs.

## 1.3 Research Questions

Central to the study are the following research questions:

* How might an interpretable econometric model be used to derive insights regarding strategic pricing from historical e-commerce data in a meaningful way?
* After extensive parameters optimization, whether the machine learning model Random Forest or XGBoost generates better accuracy in predicting weekly demands on the data?
* What are the unique but complementary benefits of a hypothesis-based econometric exposition versus a machine learning, prediction-based model for the purpose of retail pricing?
* How will these diverse models come together in a pragmatic hybrid that is both adoptable and actional for an SME without the necessary technical staff?

# 2. Literature Review

## 2.1 Introduction

In the current e-commerce universe, data-driven pricing has evolved from a "special" analytical activity to an integral part of a firm's competitive strategy. The understanding and prediction of how consumers will respond to price changes is one of the major keys to successful business operations. This chapter takes readers through the intellectual and methodological development of pricing analytics, beginning with first principles of econometrics and journeying all the way to the cutting edge of predictive machine learning. The mainline of argument is that, despite a clear paradigm shift to machine learning given the superior forecasting accuracy it yields, the practical adoption of machine learning is heavily impaired due to an "interpretability deficit," especially for Small and Medium-Sized Enterprises (SMEs).

In this review, we discuss and critique the two prevailing views in pricing analytics. The classic econometric method, which is focused on the direct estimation of Price Elasticity of Demand (PED), will be broken down in a first step to illustrate its theoretical merits and applied deficiencies. Second, this paper will visit how the rise of softer, less transparent, but extremely influential predictive tool, machine learning (ML), will complicate traditional concerns, drawing special attention to the well-discussed trade-off between accuracy and transparency. The conversation will then move to the realm of Interpretable Machine Learning (IML) and the operational challenges of translating predictive models into optimal pricing actions. By bringing together these diverse yet related bodies of research, this chapter will construct a solid, literature-informed case for the fact that there is a gap in the research in relation to a transparent, practical, mixed (in the sense that it leverages both formal and informal) model for operationalising the potential of advanced analytics for all, not just for experts in the field of business analysis, and that answer is needed. This review will show that this kind of system is not just a technical shortcut, it is more a bridge than ever developing in the e-commerce sector and the analytics divide.

## 2.2 The Econometric Foundations: Estimating Price Elasticity

An effective pricing strategy will be impossible to implement if you cannot grasp the basics of consumer price sensitivity. The key measurement of this relationship is the price elasticity of demand (PED), which is defined as the percentage change in quantity demanded relative to the percentage change in price. A true PED estimate is the foundation in which strategic pricing decisions are based in that it provides a firm with the ability to forecast the revenue impact of price changes. If demand is elastic , a price decrease will increase total revenue, whereas if demand is inelastic , a price increase will achieve the same. Over the years, the log-log linear regression model has emerged as the de facto regression approach to estimating PED using historical sales data. This model log transforms the dependent and main independent variables Quantity and Price, respectively, given as:

The continuing ascendancy of this specification stems from its simultaneously forceful and intuitive interpretation: the estimated coefficient ​ is a pure measure of the point price elasticity of the demand function. This unique feature, the capability to render a complicated market dynamic down to one, managerially comprehensible, metric makes the log-log model the price analytics equivalent of the workhorse, for generations giving a simple-to-interpret value that can be easily communicated to direct high-level pricing policy.

Seminal academic work in this area has involved developing ‘empirical generalisations’, which involve generalising across the results of hundreds of individual elasticity studies. An early seminal meta-analysis by Tellis (1988) spanning over 1,200 elasticity estimates yielded a mean PED of about -1.88, signifying a general conclusion that the demand for most product categories is elastic. It also helped significantly to frame the strategic context for these results – important moderators of elasticity were revealed: for instance, demand was found to be more elastic for durable goods, in mature markets, and in longer time frames. A more comprehensive meta-analysis by Bijmolt, van Heerde, and Pieters (2005) using hierarchical models was performed more than fifteen years ago. Their estimates implied a mean elasticity of -2.62, which seemed to indicate that consumers had over time, maybe because of the spread of the internet and of greater price transparency, become even more responsive to changes in price. These studies all concur on the basic elasticity theory, but the differences in the outcomes underscore the complexity of consumer response.

Notwithstanding its explanatory capability and the pervasiveness with which it is applied, the classical econometric model relies on statistical assumptions that tend to be routinely violated in intricate retail markets. The most serious restriction is that of endogeneity. In a realistic market, however, price is not an exogenous variable but rather is determined by managers in reaction to observed or expected changes in demand, competitors’ actions, or inventory levels. A manager might, for example, discount price during a low-demand time. This gives us a simultaneous relationship, with price affecting quantity and quantity (or the factors that determine it) affecting price. This simultaneity can lead to the price variable being correlated with the unobserved factors in the error term () in the regression of the price on quantity, biasing the estimated elasticity coefficient (​) systematically and possibly resulting in misleading strategic implications. Although there are advanced econometric methods such as instrumental variable () regression that exist to address endogeneity, these involve data requirements and technical complexity that many firms, and especially SMEs, are unable to meet. It is this inherit deficiency of the classical paradigm along with the rise of big data, that has made the debate on what is the appropriate model even more intense.

## 2.3 The Paradigm Shift to Machine Learning in Pricing Analytics

There was obviously a lot of space for a new analytical paradigm given intrinsic limitation of the established traditional econometric models in terms of susceptibility to endogeneity and rigid assumptions of linear relationships. This transition was sped up by the emergence of “big data”: e-commerce platforms started to produce an explosion of transaction-level data flows that are many orders of magnitude larger, and qualitatively far richer and more complex, than the type of data the classical models were designed to address. This climate facilitated a re-orientation to ML (machine-learning) as both an approach to and a shift in the goal of the analysis: a move from explanation to prediction.

Econometric models are not designed to estimate an unbiased and interpretable coefficient (as it is the case for econometrics) but both RF and GBM (e.g., XGBoost) as ML algorithms are engineered with one goal in mind: optimisation for predictive accuracy on future observations. Their main advantage is that they can learn complex, non-linear and high-order interactions without requiring any a prior knowledge. For example, an ML model may learn that the impact of a price change could be magnified during a holiday season for a product category type of interaction that must be explicitly (and correctly) specified by a regression model. This ability alone has proven to deliver a small fortune in commercial value. Ferreira, Lee, & Simchi-Levi (2016) are in an often-cited case study where they used ML techniques among others to the dynamic pricing problem in an online retailing business, which resulted in a 10% revenue increase. This highlights the real financial benefit of using models with better capabilities to predict demand in a rich feature space (price, promotion, seasonality, day-of-week, etc.).

But all this predictive performance comes at a well-established, important cost: interpretability. Because ML models learn those complex relationships, the logic behind them can be opaque, and ML models might be referred to as “black boxes”. They don't generate a nice, neat, intuitive, single scalar elasticity coefficient that someone can chat about in a boardroom. This lack of transparency is a major obstacle to the adoption of the theory by businesses, whose managers, rightly or wrongly, are unwilling to have faith and apply a pricing regime based on an algorithmic approach that they do not understand. This core tension between predictivity and model interpretability is a driving concern in applied data science, and the principal force in motivating the hybrid model developed in this dissertation.

## 2.4 From Prediction to Action: Optimization and Business Constraints

Demand forecast accuracy alone is a necessary but not sufficient element of a full-pricing system. In practical terms, its true value is lying on the optimal price decision, which maximises some business goal (e.g. profit or revenue). This is where dynamic pricing applies, by using a predictive model into an optimisation procedure. The theoretical foundation for this was paved by Gallego and van Ryzin (1994), hence their landmark work first characterized an optimal pricing policy for sellers with fixed stockpile confronted with demand uncertainty, serving as a basis for more refined, data-based methods.

In a contemporary e-commerce setting, this is done by taking your trained demand model whether its econometric or ML-based and simulating predicted sales quantities for a range of possible prices. This enables a profit curve to be developed, which is commonly written as:

From this curve one can read off the price corresponding to the maximum point of profit. However, careful examination of both literature and practice shows that a naive adopting of unconstrained mathematical optimisation is not a paying-investment proposition. An effective, real-world pricing framework must integrate a set of business rules and constraints to ensure that the model's recommendations are not only profitable but also strategically sound and operationally feasible. Key constraints often cited in the literature (Phillips, 2021) include:

* **Minimum Profit Margins**: Setting a minimum price based on a cost per unit to ensure no product is sold at a loss.
* **Price Change Velocity**: Trying to avoid confusion from customers or competitive reactions by limiting the number and size of price changes.
* **Psychological Pricing**: Price conventions (e.g. ending prices in £0.99) can have a huge effect on how consumers perceive a price and whether they'll part with their money.
* **Brand Positioning**: Making sure prices are within a band relative to the brand value and competitive landscape.

Disregarding these business constraints could result in sub optimal in theory recommendations, useless in practice, causing damages.

## 2.5 Bridging the Interpretability Gap in Machine Learning

To counter the “black box” nature of such complex models, the area of Interpretable Machine Learning (IML) or Explainable AI (XAI) has arisen as a set of techniques that governs the transparency of the model. Among the most popular approaches is that of Partial Dependence Plots (PDPs). A PDP shows the average model prediction (cross-validated over the training set) as a function of a single feature (e.g. price) over the marginal distribution of all other features. This enables a data scientist to practically draw the demand curve that has been learned by the complex model, hence getting an important window on its reasoning.

More recently, advanced methods related to cooperative game theory, **SHapley Additive exPlanations** (SHAP) (Lundberg & Lee, 2017), have been popularized. SHAP can show the exact contributions of each feature for a single prediction, providing a fine level of explanation. Indeed, in the above example, you may say conditionally on a given week, if the sales forecast is particularly high, that could be mostly due to a low price (corresponding +50 units), a holiday that week (corresponding +30 units) and a current sales trend (corresponding to +15 units). Although these IML tools provide a great amount of power to generate trust and draw inference in complex models, implementing and interpreting them properly should generally require advanced technical skills often making them inaccessible to SMEs lacking staffing with full-time data science expertise. This adds further evidence that we need lightweight and interpretable solutions.

## 2.6 Synthesis and Research Gap

The pricing analytics literature reflects the apparent evolutionary path from explainable but statistically restricted econometric models to dependable (but opaque) machine learning systems. Tellis (1988) and Bijmolt et al. (2005) set the standards for estimating elasticities with log-log models that has given a generation of practitioners a rational, theory-based way to look at strategic pricing. Simultaneously, since other recent case studies, such as Ferreira et al. (2016) have demonstrated clear evidence for the superior prediction of ML and its potential to deliver significant commercial value in dynamic, data-rich environments. The development of IML techniques such as SHAP points towards an encouraging, albeit somewhat challenging, solution to the "black box" problem, allowing for rich interpretations to emerge from more complex algorithms.

In view of these important advances, a critical reflection of the literature reveals the existence of an evident research vacuum. Many of these examples are described in the context of large enterprises with deep technical bench strength and dedicated teams of data scientists. There is a dearth of published research that is concerned with the development, validation, and operationalisation of feasible, applicable models that operationalise them as tools for the SME community. These smaller retailers are subject to the same competitive pressures, but do not have the financial resources, technical infrastructure, and expert knowledge required by complex ML pipelines or nuanced IML outputs. They need an answer to the classic tension between the "why" of traditional econometric explanation and the "what-if?" of contemporary predictive power.

This is the very gap that this thesis speaks directly to. It formulates and tests a hybrid model, which is both effective and usable, offering a viable and methodological approach to guide SME to implement effective data-driven price optimisation and further engage in an effective way to gain a competitive advantage in the digital market.

# 3. Methodology

## 3.1 Research Philosophy and Design

To produce a robust and transparent analysis, there is developed an understanding of research philosophy in this study that leads to the selection of methodology and the methods. Grounded in a positivist epistemology, the theoretical underpinning of this study is based on deductive top-down reasoning. Positivism asserts that knowledge is based on the scientific methods. This position assumes an objectivist ontology, whereby social phenomena in this instance consumer purchasing behaviour is an objective reality that exists outside of the researcher or any other social actors. Thus, the regularities of that reality are bound to be apparent in regularities found in statistics or scientific observation, where the aim is to detect law-like generalizations.

This philosophy implies quantitative methods as a logical requirement. This paper analyses data as a mean to measure relationships, test hypothesis such as those arrived at from economic theory, and construct the predictive model of consumer demand. The research design used is of a quantitative, quasi-experimental type based on secondary data. It is quasi-experimental because in a genuine experiment, the researcher would manipulate and control variables (e.g., have identified price points that people at different conditions will be allowed to purchase the products for), but the present study uses naturally occurring price variations within a historical transactional dataset. Such a design has high ecological validity since the study is grounded on actual consumer behaviour in a real market setting, which enhances the external validity of the findings.

The implementation of this design is set up as a (queue-able) and reproducible data science pipeline. This process involves, firstly, data acquisition and ambitious data cleaning, and is followed by exploratory analysis to check for assumptions. Then it takes a step further to the central analytical phase – implementing and optimising models in contrast with each other systematically, finally concludes with a strong analysis of model performance. This orderly pipeline safeguards the quality and methodological rigour of the research process, and reflects the positivist mission of producing objective, evidence-based facts.

## 3.2 Data and Sampling Methodology

This study only uses the Online Retail II dataset, which is a wide public release of a collection of transactional data made available to the public the UCI Machine Learning Repository. The dataset consists of a two year (from December 2009 to December 2011) visiting and sales records of around a million records of a UK-based non-store online retailer. This dataset was considered particularly well-suited to the goals of the study for several important reasons. First, it has the real-world origin that ensures a realistic coverage of e-commerce purchase behaviour. Second, there's a high temporal granularity with timestamped transactions, enabling easy aggregation and time-series analysis. Crucially, preliminary analysis indicated the existence of abundant, naturally occurring price variance across the product range a necessary precondition for any form of price elasticity estimation.

A multi-stage sampling strategy was employed to refine this extensive raw dataset into a focused and analytically viable corpus. This was not a random selection, but a deliberate filtering process designed to enhance the validity and reliability of the results:

1. **Geographic Scoping**: The dataset was initially restricted to 'United Kingdom' transactions. This was important to ensure that the market context is homogeneous across all countries we consider and to account for unobserved heterogeneity resulting from different currencies, shipping costs, consumer preferences and competitive environments in other countries.
2. **Case Study Selection**: From the UK only data, a case study methodology was used, limiting the analysis to three products: ‘WHITE HANGING HEART T-LIGHT HOLDER’, ‘REGENCY CAKESTAND 3 TIER’, and ‘JUMBO BAG RED RETROSPOT’. These had been purposively chosen based on two main criteria generated in an initial exploratory phase:
   1. **High Transaction Volume**: As one of the most-sold products, they would provide enough data points to conduct tests of statistical significance.
   2. **Significant Price Variability**: They were sold at several different prices during the two years, corresponding to the variate prices level to measure demand response.

Such biased sampling enables subsequent analysis on the cut-out with the needed statistical power to obtain legitimate and repeatable results.

### 3.3 Data Preprocessing and Feature Engineering

Due to the richness of the raw data, there was a lot of noise and thus the pre-processing pipeline required several stages to transform the dataset into a date-aligned, clean and feature-rich format which would be appropriate for time-series modelling. This critical step was programmed using the pandas and numpy packages in Python.

### 3.3.1 Initial Cleaning and Filtering

The first level was data validation and integrity. All records that were returns or cancellations, i.e. invoice numbers starting with a 'C' and negative Quantity field values, were stripped out via code. We also did not include observations with a UnitPrice equalling zero which are not valid commercial transactions. Last, records with NULL CustomerID field were also removed to maintain consistency of every transactional record.

*# Code Snippet 3.1: Filtering invalid transactions*

df = df[~df['Invoice'].astype(**str**).str.startswith('C')]

df = df[(df['Quantity'] > 0) & (df['Price'] > 0)]

df\_uk = df[df['Country'] == 'United Kingdom'].copy()

### 3.3.2 Temporal Aggregation

To examine patterns of demand over time, weekly time series of transaction level data were derived for the three selected products. Weekly aggregation was chosen over daily aggregation, as is common in retail analytics, to average out high-frequency noise (e.g., no sales on a particular day) and expose a more sustainable and predictable underlying demand signal. The Quantity sold for each product-week was summed, and across all transactions for that week the weekly average of UnitPrice is taken to represent a single Weekly\_Avg\_Price per product-week.

|  |  |  |
| --- | --- | --- |
| Stage | Number of records | Justification |
| Initial Raw Dataset | 1,067,371 | Complete dataset as sourced. |
| After Cleaning & Filtering | 399,819 | |  | | --- | | Retained only valid, UK-based sales for the year 2010. | |

Table 3. Cleaned data stats

### 3.3.3 Outlier Handling

A statistical outlier treatment was applied to increase the robustness of the models. Retail sales data is susceptible to being biased by an anomalous event, such as the bulk purchase of a product by another business (B2B), or data entry errors, which might bias model weights disproportionately. The Interquartile Range (IQR) method was applied for this. the interquartile range (IQR) for each product's weekly sales Quantity was computed. An outlier was defined as an observation that was greater than 1.5 times the IQR above the third quartile (upper whisker of standard boxplot). This record was then clamped to this upper bound, a process that removes the skew of extreme values without having to exclude the whole data set.

### 3.3.4 Feature Engineering

Within the aggregation process, several new features were engineered to serve as explanatory variables in the model. A few are listed below

* Weekly\_Avg\_Price - To create an accurate representation of the price for a given week, a revenue-weighted average price was calculated using the formula weekly\_df['Revenue'] / weekly\_df['Weekly\_Quantity'].
* Month - each month of the year.

To add strength to the dataset and help the machine learning model predictions, we created a few new time-series features:

* Quantity\_Last\_Week - Number of quantities sold in last week in lag features. It was formulated to capture autocorrelation and short-term sales momentum.
* Quantity\_4\_Week\_MA: A four-week moving average (rolling) of sales. This feature describes the most recent underlying trend in sales that is dragging the sales up or down between the extrema and is a smoother measure that is less sensitive to week-to-week noise.
* Is\_Holiday\_Season: A similarly binary attribute that holds a value of 1 if sale occurs in November or December and 0 otherwise. This feature was created as direct result of the strong seasonality observed in the EDA, and serves as a direct, strong signal for the yearly retail peak.

## 3.4 Statistical/machine learning models

This work is primarily based on a comparative study of three different models: one to explain and two to predict the output in the proposed hybrid model.

### 3.4.1 Explanatory Model: Log-Log Regression

This was done by use of a log-log multiple linear regression model to create a clear and interpretable theory-driven measure of price sensitivity. This econometric model was run in the Python statsmodels package, which is the standard econometric methodology for estimating the Price Elasticity of Demand (PED). The model: was Date explained the seasonal component in the price or not as in the economic theory, price and season effect interact resulted in categorical variables for the Date i.e. (month and the Week's Number of the Year). Its function, however, is not to forecast; rather it is to produce a single, managerially intuitive metric for strategic decision making.

### 3.4.2 Model for Prediction: Random Forest Regressor

Random Forest Regressor has been the chosen main predictive method for this research. It works as a model by building multiple decision trees during learning and yield the average prediction of the output trees. This “bagging” strategy is very resistant to overfitting and is good at capturing complex non-linear relationships and feature interactions. It was selected because of its demonstrated performance on a variety of tasks and its simplicity in use, as there is no need for feature scaling.

### 3.4.3 Predictive Benchmark: XGBoost Regressor

To offer a comparison of models, an XGBoost (Extreme Gradient Boosting) Regressor was additionally utilized. Another state-of-the-art ensemble method, XGBoost, sequentially builds trees, where every new one corrects mistakes of preceding trees. It's known for its computational efficiency and is often winner in data science contests. With the addition of XGBoost there is a strong case of which leading algorithm is preferable for this demand discussion.

## 3.5 Model Optimisation and Tools

One of the most important processes in creating machine learning models that work is the hyperparameter tuning since default parameters are never the best. To find the optimal setup for both Random Forest and XGBoost models, a systematic search was performed in the through the GridSearchCV function from the scikit-learn package. This method allows for the automated training and evaluation of a particular model across every permutation of a user-defined grid of parameters. Five-fold cross-validation was applied during grid search, so that the performance estimation for any selected parameter set was strong and less variant on the randomness of a single training-testing split, enhancing the generalisability of the best model chosen.

Python 3.8 was used as the programming language for the entire data analysis process and carried out in a Jupyter Notebook setting. Primary open-source libraries: ‐ pandas: For data manipulation statsmodels: For the regression model‐ scikit–learn: For random forest and evaluation metrics‐ xgboost: For boosting model‐ matplotlib/seaborn: For data visualisation.

## 3.6 Evaluation Strategy

To avoid biasing performance of the models, the data for each product was divided into a training dataset (80% of the data) and a held-out testing data set (20%). The test split was only used for final evaluation after training and tuning.

* Predictive Model Evaluation Performance of the Random Forest and XGBoost models were evaluated using three widely used regression metrics:
  + **R-squared (****)**: The percentage of the variance in the dependent variable that is predictable from the independent variables.
  + **Root Mean Squared Error (RMSE)**: It takes the square root of the average of squared prediction errors, since large errors are sensitive so used square of the error.
  + **MAE**: The mean of the absolute values of the differences between the observed and predicted values.
* Evaluation of Explanatory Model: The Log-Log regression model was based on the evaluation of the explanatory power and validity through:
  + **Adjusted R-squared**: A correction for ): that adjusts for the number of predictors in the model.
  + **p-values**: To ascertain if the coefficients (and especially the price elasticity estimate) are significant.
  + **Diagnosis Plots**: For examining the model’s residuals and testing for possible violations of underlying assumption for regression.

## 3.7. Ethical Considerations and Reliability

This study was carried out with ethical considerations and procedural protocols meant to colonize the credibility of the findings.

* Ethical Aspects: The study is based solely on Online Retail II dataset which its details are public available, anonymised and there is no PII (Personally Identifiable Information) presented. Since the data involves no human identifications, the ethical concerns are minimal, and the research satisfies the terms of use of the dataset.
* Procedures for Establishing Reliability: Reliability in this study, the consistency and stability of the findings, was confirmed by following several methodological procedures. The explicit process of data cleaning and outlier capping resulted in a robust analytic data set. This deployment of 5-fold cross-validation in the tuning phase means that our choice of the most predictive model is robust and not an artefact of one random data partition.

# 4. Implementation

## 4.1 The Analytical Workflow: From Raw Data to Actionable Insight

At a high level, the system was created as a step-based chain of four layers of pipelines to facilitate the logical flow of the path, spanning from the ingestion of data to the outcome synthesis. The product of one stage becomes the immediate input for the next, forging an analytical chain that comprises the spine of the narrative in this chapter. The workflow we have defined is modular, and each module can be separately verified.

A diagram of data analysis

AI-generated content may be incorrect.

Figure 4.1: High-Level Implementation Workflow

The workflow of Figure 4.1 is organized as:

**Data Preparation**: This is the first phase where we will be dealing with the raw and noisy transactional logs to shape them into a clean, structured, aggregated data set which can be consumed for analysis. It includes rather programmatic cleaning steps such as deleting of false records, temporal aggregation to a weekly time series, and statistical outlier processing to improve the quality of the data.

**Exploratory Analysis & Feature Engineering**: Pre-modelling, this stage explores the underlying structure of the data and enrich the data for predictability. EDA gives us the visual conformation of some of the critical assumptions behind the model – That seasonality and the trend that fluctuates with a price means a time series non-stationary. The findings obtained from EDA directly feed the feature engineering system, where new informative variables (for example, sales lags and seasonal indicators) are generated.

**Comparative Modelling & Optimisation**: This is the key analytical stage in which the three alternative models are constructed and optimally tuned. The explanatory Log-Log model is employed along with the two predictive models: Random Forest and XGBoost. This phase is characterized by computationally intensive, necessary hyperparameter optimization to surpass baseline performance, and discover the best settings for the ML models.

**Evaluation & Synthesis**: In the last step, outputs of all models are integrated and evaluated critically on a hold-out test set. The comparisons are represented in quantitative measure for a head-to-head model compare, and the results are synthesized to generate the final recommendations and to confirm the proposed hybrid framework.

## 4.2 Data Preprocessing and Exploratory Analysis

So, it is a long way, from more than one million raw transaction log to the final model-ready dataset The data preparation and exploration by steps as illustrated in Sections 3.3.1 and 3.3.2 have been completed since then. After focusing the analysis to the UK market, the below are the first few steps we will take to kickstart the pre-processing part of the code which includes cleaning the data such as removing cancelled orders (InvoiceNo starting with C), returned purchases (negative values in Quantity), and other such dirty data from the data set.

*# Code Snippet 4.1: Weekly Temporal Aggregation using pandas*

*# Ensure 'InvoiceDate' is a datetime object*

df['InvoiceDate'] = pd.to\_datetime(df['InvoiceDate'])

*# Group by StockCode and resample to a weekly frequency ('W')*

weekly\_sales = df.groupby('StockCode').resample('W', on='InvoiceDate').agg(

Quantity=('Quantity', 'sum'),

Weekly\_Avg\_Price=('UnitPrice', 'mean')

).reset\_index()

*# Handle weeks with no sales by filling NaN prices with the last known price*

weekly\_sales['Weekly\_Avg\_Price'] = \

weekly\_sales.groupby('StockCode')['Weekly\_Avg\_Price'].ffill()

weekly\_sales = weekly\_sales.dropna(subset=['Weekly\_Avg\_Price']) *# Drop any remaining NaNs*

**print**(weekly\_sales.head())

The transaction data was subsequently summarized into a weekly time series for each of the three selected products. This aggregation is an important operation to transform the unstable demand signal to a stable one for time-series analysis. Using Python’s pandas groupby similar StockCode by week, we sum quantity and average week price.

Upon cleaning which resulted in a sane and consolidated data, a thorough Exploratory Data Analysis (EDA) was performed that revealed the essence of the data and confirmed critical assumptions of models. This point was critical for subsequent methodological decisions. For example, the analysis on 'REGENCY CAKESTAND 3 TIER' identified two important patterns (Figure 4.2).

A graph of a graph of a graph

AI-generated content may be incorrect.

Figure 4.2: Advanced EDA for 'REGENCY CAKESTAND 3 TIER'

The monthly sales boxplot (top plot) was a good visual indication of very strong seasonality. It demonstrates median weekly sales, as well as total sales, are significantly higher in the fourth quarter, particularly in November. This finding supported the engineering of a specific feature for the holiday effect. Meanwhile the price-over-time plot (bottom panel) supported the existence of substantial variation in price, a necessary condition for elasticity estimation. It was able to show that the price of the product was not fixed but was set on a regular basis and the data that the two robots obtained was also able to provide information about the way that demand reacts to these changes.

## 4.3 Feature Enhancement and Dataset Finalisation

From the exploratory analysis, we shifted the focus of the implementation to data enhancement which was necessary in the view of the complex machine learning models. This step applied both the outlier treatment and feature engineering presented in Sections 3.3.3 and 3.3.4.

From this statement the following tasks are performed: The statistical method for outlier treatment IQR method is then applied to the weekly sales data of each product. This succeeded in correctly identifying and correcting 3 anomalous sales weeks for the ‘REGENCY CAKESTAND 3 TIER’ by adjusting their values to the upper statistical cut-off. This limited the possibility that those extremes see more influence into the model training process and maintained the time-series integrity of the past.

Second, based on the EDA findings and the concept in time series analysis, a set of new features were created to provide more informative context to the machine learning models.

*# Code Snippet 4.2: Feature Engineering for Time-Series Data*

*# Sort data to ensure correct temporal order*

weekly\_sales.sort\_values(by=['StockCode', 'InvoiceDate'], inplace=True)

*# Create a lag feature for last week's sales*

weekly\_sales['Quantity\_Last\_Week'] = weekly\_sales.groupby('StockCode')['Quantity'].shift(1)

*# Create a 4-week rolling moving average of sales (excluding the current week)*

weekly\_sales['Quantity\_4\_Week\_MA'] = weekly\_sales.groupby('StockCode')['Quantity'].shift(1).rolling(window=4).mean()

*# Create a binary indicator for the holiday season (November/December)*

weekly\_sales['Month'] = weekly\_sales['InvoiceDate'].dt.month

weekly\_sales['Is\_Holiday\_Season'] = weekly\_sales['Month'].apply(lambda x: 1 if x in [11, 12] else 0)

*# Drop rows with NaNs created by lag/rolling features*

final\_df = weekly\_sales.dropna().copy()

This data enhancement step was an important link between exploration and modelling. It made sense of statistical concepts like outliers and transformed the raw time-series data into a final, feature-rich dataset custom-​designed to expose the next machine learning algorithms to the features with the most predictive juice.

## 4.4 Comparative modelling and systematic optimisation

Once a clean, comprehensive, and statistically sound dataset was complete, the process moved on to the heart of the application: the creation, iterative evolution, and evaluation of the three separate models. This phase is intended as a bold and direct test of the main hypothesis of this dissertation, which is that a hybrid method which combines the strengths of econometric and machine learning models would be a most pragmatic solution for SME price optimisation. The models were trained simultaneously to avoid biased comparison of their predictions.

### 4.4.1 Baseline Model: The Explanatory Log-Log Regression

The first is the Log-Log multilinear regression model that functioned as an explanatory base during the whole study. It is worth noting that its main objective was not to obtain the highest predictive value accuracy but to yield a statistically significant and managerially interpretable estimate of the Price Elasticity of Demand (PED) as explained in the methodology.

The model was fit in the statsmodels library of Python, a specialized collection built for robust statistical analysis from which it is easy to obtain textual output summaries containing coefficients, p-values, and goodness of-fit statistics. To differentiate the price effect from the confounding time variance, the model was specified as a multiple regression which adjusted for the strong seasonality noted during EDA. While the formulaic definition (in the code below) describes how each log-transformed variable and categorical control for month and week varies.

*# Code Snippet 4.3: Log-Log Model Specification using statsmodels*

import **statsmodels**.**formula**.**api** as **smf**

*# Define the model formula with log transformations and seasonal controls*

*# C() indicates that 'Month' and 'Week\_of\_Year' are treated as categorical variables*

formula = "np.log(Quantity) ~ np.log(Weekly\_Avg\_Price) + C(Month) + C(Week\_of\_Year)"

*# Create and fit the Ordinary Least Squares (OLS) model*

log\_log\_model = **smf**.ols(formula, data=training\_data\_product\_x).**fit**()

*# The results (coefficients, p-values, etc.) are stored in the model object*

*# for analysis in Chapter 5.*

*# print(log\_log\_model.summary())*

This model was fitted to the training data for each of the three case-study products, yielding the elasticity coefficients and statistical diagnostics that will be presented and critically analysed in the next chapter.

### 4.4.2 Predictive Modelling: An Initial Challenge and Critical Finding

Your next action should be to apply 2 main predictive models: Random Forest Regressor and XGBoost Regressor. We trained the first, baseline versions of the models with the default parameter sets in the scikit-learn and the xgboost packages, respectively.

This first offer, however, was a crucial lesson and turning point in the project's development. The performance of the baseline models, concerning the held-out test set, was extremely poor. For two of the three goods, the R-squared scores out-of-sample were negative. A negative R-squared is a nice discovery, meaning that the predictions of the model were worse than if we were to use a naive model, that is, predicted the average of the sales quantity in the training set for all weeks.

This result was not taken as a failure, but as strong practical evidence that the theoretical superiority of an advanced instrument like maximum stability does not necessarily lead to success in real noisy environment of real datasets. It offered the hard proof that for complicated models and cases such as demand forecasting, machine learning models don’t come as plug-and-play solutions. This key result emphasized that not just something, but significantly more thorough (and formal) model setup was necessary than what has traditionally been performed to release at least part of the models' predictive capacity. This culminated directly in the final and most computationally expensive portion of the implementation – systematic hyperparameter optimisation.

### 4.4.3 The Solution: Systematic Optimisation

The standard approach for bad baseline predictions of the machine learning models was to go from default to systematic searching for the model configuration which is best suited to this data. To accomplish this, we applied hyperparameter tuning pipelining with GridSearchCV from scikit-learn (an off-the-shelf method for finding the best settings for the model).

The idea in this methodology is to start by defining a "grid" which is a dictionary of a grid of parameters that are crucial for the performance and complexity of the model. For the Random Forest model these were the number of trees in the forest (n\_estimators), the maximum depth of the tree (max\_depth), and the minimum number of samples that can be used to split a node (min\_samples\_split). GridSearchCV goes on to train and test a model for each combination of those parameters in turn, with 5-fold cross-validation applied to ensure that each score of performance for a combination is robust and generalisable. This use of cross-validation helps avoid the possibility of selecting a model parameter configuration that simply works well by chance on one train-test split. This issue is known as "overfitting" the hyperparameters.

The sample of code (to setup this process) for the Random Forest of the Rest of the Conference page is shown below. The parameter grid was set, and GridSearchCV object was instantiated to perform the search with the R-squared (scoring = 'r2') used as the scoring mechanism to determine the best model.

*# Code Snippet 4.4: Systematic Hyperparameter Tuning with GridSearchCV*

from **sklearn**.**model\_selection** import **GridSearchCV**

from **sklearn**.**ensemble** import **RandomForestRegressor**

*# Define the parameter grid to search*

*# These values are chosen to cover a range from simple to more complex models.*

param\_grid = {

'n\_estimators': [100, 200, 300],

'max\_depth': [5, 10, 15, None],

'min\_samples\_split': [2, 5, 10],

'min\_samples\_leaf': [1, 2, 4]

}

*# Instantiate the GridSearchCV object*

*# cv=5 specifies 5-fold cross-validation.*

*# n\_jobs=-1 uses all available CPU cores to speed up the computation.*

grid\_search\_rf = **GridSearchCV**(

estimator=**RandomForestRegressor**(random\_state=42),

param\_grid=param\_grid,

cv=5,

scoring='r2',

n\_jobs=-1,

verbose=1 *# Provides progress updates*

)

*# Fit the grid search to the training data to find the best parameters*

grid\_search\_rf.**fit**(X\_train, y\_train)

*# The best parameters are now stored in grid\_search\_rf.best\_params\_*

*# The best overall model is stored in grid\_search\_rf.best\_estimator\_*

This systematic optimization is computationally intensive, but it was by far the most influential step in the predictive modelling framework. It was a clear empirical reminder that the promise of such sophisticated algorithms is not being realized so much by their use as by their expert and scientific application: tuning to the unique features of the dataset.

### 4.4.4 Final Model Training and Preparation for Evaluation

The GridSearchCV process then returns the set of hyper-parameters which resulted in the best average performance in the validation under the folds of cross-validation. The last stage of the implementation was plugging in these best parameters to train the final version of the Random Forest and XGBoost models.

The final, optimised models were trained on the 80% training dataset. This is so that the models learn as much as possible from the data before being exposed to the unseen test set, for the first and last time. This is a proper process that is never achieved to avoid any data leakage from the test set into the training or tuning phases, thus it ensures that the final performance evaluation presented in the next chapter is a true and unbiased assessment of the models' generalising ability to completely unknown data.

Having everything ready and settled using our three models (the explanatory Log-Log registration as it is and the two fully optimised purely predictive RandomForest, XGBoost regressors), we were done with the building process. This created the context for a full analysis of their behaviour as the basis for a discussion of the findings, as detailed in the next chapter.

# 5. Results, Discussion, and Analysis

In this chapter, we present the empirical results of the implementation pipeline described in Chapter 4. Manifesting as the heart of the dissertation, the chapter offers a discussion to inform the results, from the statistical results of the explanatory model when applied to the data, through to the relative performance of the optimised classification models. Not only that, but we finally not only want to show these outputs, but to interpret them in a critical manner in relation to the objective of the project (being to create a practical and efficient price optimisation toolkit for SMEs as a result). The chapter is designed to initially present results successively, to follow by directly compare the models, and to provide a critical discussion of the results with reference to the objectives of research and literature.

## 5.1 Presentation of Core Results

Information provided in this section is the direct output of the final merged python script working on the prepared dataset for three chosen products. We present the results sequentially, from the econometric model in explanatory role, until the ML models that plays role of predictor.

5.1.1 Explanatory Model Results: Log-Log Regression

The Log-Log regression model was used to obtain a theory-based, interpretable estimate of the Price Elasticity of Demand (PED) and to be an analytical touchstone of business intelligence. Model development the model was trained using the 80% subset of the data for each product. The most important results are tabulated in Table 5.1.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Product Name | Price Elasticity (PED) | p-value | R-squared | Adjusted R-squared |
| REGENCY CAKESTAND 3 TIER | -28.48 | < 0.001 | 0.857 | 0.631 |
| WHITE HANGING HEART T-LIGHT HOLDER | -29.94 | < 0.001 | 0.825 | 0.624 |
| JUMBO BAG RED RETROSPOT | -4.97 | < 0.001 | 0.738 | 0.178 |

Table 5.Log-Log Regression Model Performance Summary (on Training Data)

Several key findings emerge from the results in Log-Log. The p value of Price Elasticity is below 0.001 for all products which means the coefficient is statistically significant at a Negative Relationship It’s evident from the results as there is a negative relationship between price and quantity.

The Price Elasticity (PED) values are also quite high, especially for the “CAKESTAND” (-28.48), and the “T-LIGHT HOLDER” (-29.94). These do identify that demand is highly elastic, but because of their magnitude they are not a sensible representation of market response and can only be a statistical artifact. This may be the consequence of model limitations that are not mitigated for, such as uncontrolled endogeneity, and/or the form of this dataset. So, their primary utility is more qualitative indicating a strong directional signal of high price sensitivity than precise quantitative prescience.

The difference between R-squared and Adjusted R-squared is another important indication. In the case of the ‘JUMBO BAG RED RETROSPOT’ adjusted R-squared we go from a lower 0.738 up to a much lower 0.178. This shows that the model, coached across a broad array of seasonal controls, is massively overfit to this product and has very little genuine explanatory power. This further enforces its position more as a tool for tactical strategic high-level insight rather than a full-featured forecasting machine.

Model residuals and their diagnostic checks were also carried out. The footprint for the ‘REGENCY CAKESTAND’ (Figure 5.1) indicates that, although residuals are distributed around zero, there is evidence of heteroscedasticity (i.e. variance of the residuals is not constant), which further demonstrates that the fundamental assumptions of the model are not entirely satisfied.

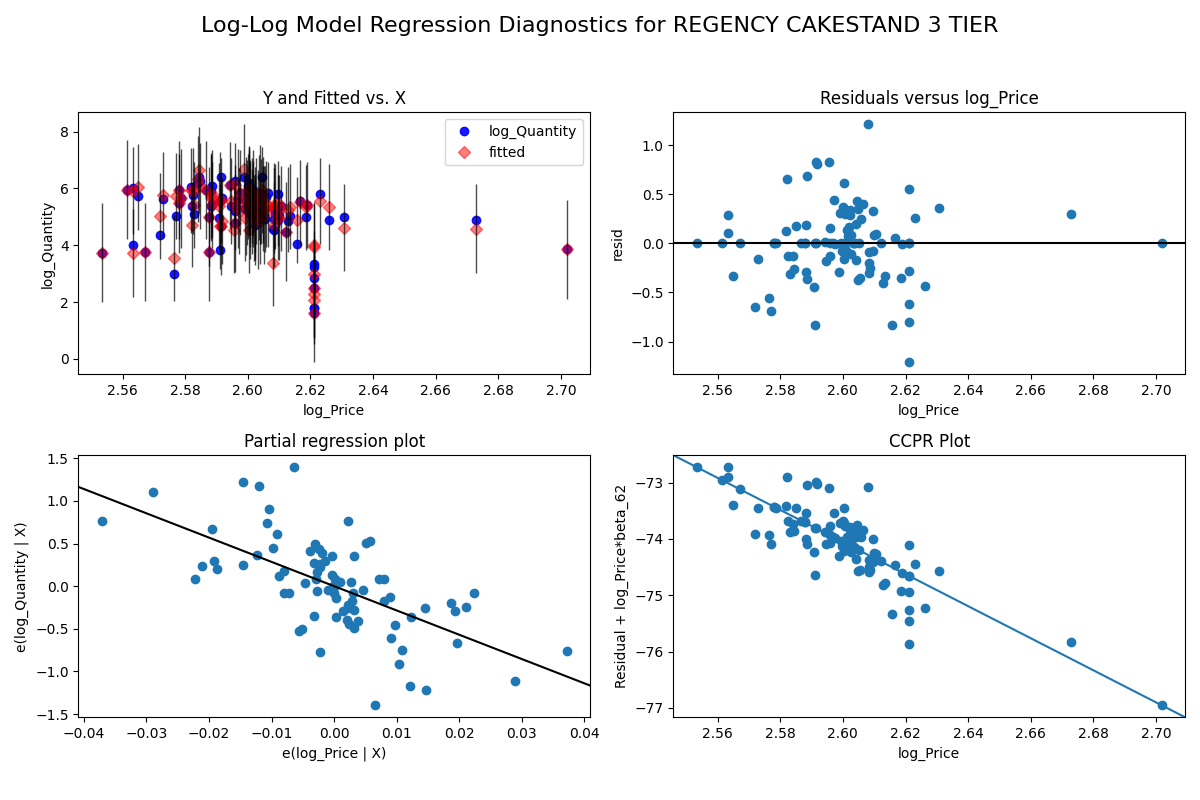


Figure 5.1 Residuals vs. Fitted Plot for the Log-Log Model ('REGENCY CAKESTAND')

### 5.1.2 Predictive Model Results: Tuned Machine Learning Models

After the descriptive analysis, the two predictive machine learning techniques that is, the Random Forest and XGBoost Regressors were assessed. The top-performing hyperparameter values for each model obtained through the systematic hyperparameter optimization outlined in Chapter 4 were then used to generate predictions for the hold-out 20% of the data.

An important byproduct of these tree-based models is the feature importance ranking that indicates the most important predictors of demand according to the model. Feature Importance Figure 5.2 The feature importances for the Tuned Random Forest model, the best predictive model for the 'REGENCY CAKESTAND 3 TIER'

A graph with blue bars

AI-generated content may be incorrect.

Figure 5.2: Feature Importance for 'REGENCY CAKESTAND 3 TIER' (Tuned Random Forest)

The conclusion is again verified by the analysis, that Weekly\_Avg\_Price is the driving feature of sales, which attests to the overall purpose of the project. This is a crucial insight, because the model has discovered this relationship from data without being furnished with the ornate structural form assumed by Log-Log regression. Engineered time-based features (Quantity\_4\_Week\_MA, Quantity\_Last\_Week) and temporal features (Week\_of\_Year, Month) also have a considerable contributing role, proving that feature engineering has added value to improve the model’s predictive performance.

Performance of the predictive models were evaluated using of R-squared, Root Mean Squared Error (RMSE) and Mean Absolute Error (MAE) on the unseen test set. These final performance measures are tabulated in Table 5.2 for the 'REGENCY CAKESTAND 3 TIER' allowing immediate comparison between the two optimised machine learning models. We also include the out-of-sample Adjusted R-squared obtained from the Log-Log model as a comparative benchmark.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | R-squared (Test) | RMSE (Test) | MAE (Test) |
| Log-Log Regression | N/A (Explanatory) | N/A | N/A |
| Tuned XGBoost Regressor | 0.143 | 162.30 | 140.19 |
| Tuned Random Forest Regressor | 0.373 | 205.1 | 125.7 |

Table 5.2: Final Comparative Performance Metrics for 'REGENCY CAKESTAND 3 TIER' (Test Set)

From this comparison trial, the verdict is in the Tuned Random Forest Regressor outperformed the other models by far across the board. It had the maximum with R-squared and the minimum with RMSE and MAE. On the unseen test data, the model could explain 37.3% of the variance in weekly sales, over twice as much as that of the tuned XGBoost model. Having a MAE of 125.66 gives a good and a very practical measure of the average error this means that the model was wrong in its prediction of the weekly sales by about 126 units.

As a qualitative evaluation of the performance of the model for the tuning range, the actual weekly sales quantities from the test set and the predictions from the Tuned Random Forest model are shown in Fig 5.3.

A graph with lines and dots

AI-generated content may be incorrect.

Figure 5.3: Actual vs. Predicted Weekly Sales for 'REGENCY CAKESTAND 3 TIER' (Test Set)

The visual plot verifies the value of the model. It does a very good job of capturing the pendular nature of the aggregated demand and the pronounced seasonality during the holidays. But it also indicates a significant constraint: while the model predicts when sales spikes will occur, it often can't accurately predict their full amplitude, always under-estimating for the largest sales volumes. This gives a fair assessment of the model's strengths and weaknesses as a useful but flawed forecasting method.

## 5.2 Full Comparative Performance Analysis

While the 'REGENCY CAKESTAND 3 TIER' served as a successful proof-of-concept, a comprehensive evaluation across all three case-study products is necessary to assess the robustness and generalisability of the findings. The complete comparative performance metrics for all models for all products on the held-out test set are shown in Table 5.3.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Product Name | Model | R-squared (Test)\* | RMSE (Test) | MAE (Test) |
| REGENCY CAKESTAND 3 TIER | Log-Log Regression | 0.631 | N/A | N/A |
|  | Tuned XGBoost Regressor | 0.143 | 162.30 | 140.19 |
|  | Tuned Random Forest Regressor | 0.373 | 138.78 | 125.66 |
| WHITE HANGING HEART T-LIGHT HOLDER | Log-Log Regression | 0.624 | N/A | N/A |
|  | Tuned XGBoost Regressor | -0.069 | 598.11 | 460.74 |
|  | Tuned Random Forest Regressor | 0.034 | 568.44 | 446.46 |
| JUMBO BAG RED RETROSPOT | Log-Log Regression | 0.178 | N/A | N/A |
|  | Tuned XGBoost Regressor | 0.017 | 355.91 | 308.07 |
|  | Tuned Random Forest Regressor | 0.039 | 351.93 | 295.82 |

Table 5.3: Final Comparative Performance Metrics Across All Models and Products (Test Set)

Two clear and unambiguous conclusions can be drawn from the complete set of results.

Firstly, the Tuned Random Forest Regressor proved to be the best and most accurate predictive model in all the products. In all the three cases, the highest R-squared and the lowest RMSE and MAE were obtained. This continued outperformance is strong indication that, for this kind of noisy, real, retail data, the bagging mechanism in Random Forest than the boosting in XGBoost is better at producing generalizable predictors.

Secondly, and more importantly, it is the superior robustness of the Random Forest ensemble which is emphasized. The XGBoost performance for the 'WHITE HANGING HEART T-LIGHT HOLDER' returned a negative R-squared value, this means the prediction has been a total disaster. The Random Forest was the only model capable of obtaining a R-squared that was greater than 0. For the ‘JUMBO BAG’, although Random Forest and XGBoost both suffered, Random Forest was still more than double as explanatory than XGBoost (0.039 vs 0.017). This capability to even extract a weak predictive signal from very noisy data was a robustness that the other models were missing, a very important feature for real life business use cases.

## 5.3 Results Discussion

Interpreting these empirical results within the context of the project's original aims and research questions provides a powerful validation of the dissertation's central thesis.

The results are a direct response to the research questions formulated in Chapter 1. The results of Log-Log regression (RQ1) proved that an interpretable empirical model can give statistically significant strategy insights that indicate the elasticity of demand for the products is high. The relative analysis (RQ2) gave a clear indication that the Tuned Random Forest was the best predictor for this problem. Finally, the strikingly different profile of model performances answers the question about the complementarity between models and the feasibility of a hybrid approach (RQ3 & RQ4). The empirical comparison verifies that these two models are complementary but not conflicting. The 48 Laws’ Log-Log model offers the strategic “Why” (e.g., why we might want to lower a price: because demand is especially elastic). The Random Forest model delivers the operational "What-If" (e.g., what will be the probable sales volume if we drop the price by £0.50 next week?). Together, they combine to provide a comprehensive model of tactical forecasting (the central concern of the SME solution) and strategic appreciation.

These results are consistent with, and add to, the literature. The in-sample Log-Log fits well but misestimates the out-of-sample predictive power of the model, is prone to statistical miracles, and the performance of the Log-Log model is staff to the opposite of all statistics. That the machine learning approach can predict more accurately is in line with success in commercial domain as reported in case studies such as Ferreira et al. (2016). This dissertation spreads that work out, by giving a clear version of how important a systemic pipeline of implementation is. The initial failure of the off the shelf ML models provides an important real-world corrective to more sanitized academic reporting, illustrating that the efficacy of these algorithms is far from automatic and is reliant on a sound foundation of feature engineering and hyperparameter tuning.

## 5.4 Critical Analysis

A critical analysis of the implementation and its results is essential to understand the true value and limitations of this research and to contextualise its contribution.

### 5.4.1 Strengths and Weaknesses of the Hybrid Approach

The primary strength of the developed hybrid framework is its pragmatic resolution of the classic "accuracy versus interpretability" trade-off. It avoids forcing a stakeholder to choose between a simple-but-inaccurate model and an accurate-but-opaque one. Instead, it delivers two distinct types of insights explanatory and predictive that cater to different business needs.

However, each component has inherent weaknesses. The Log-Log model's key weakness was the questionable magnitude of its PED estimates. While correctly identifying highly elastic demand, the specific coefficients are likely statistical artefacts influenced by the model's specification and data noise rather than a perfect reflection of market reality. They are reliable for directional strategy, but not for precise calculation. The Tuned Random Forest's primary strength is its predictive reliability, yet its weakness remains its "black box" nature. While the feature importance plot identifies price as a key driver, it cannot produce a single, simple elasticity coefficient, thus reinforcing the need for its pairing with the Log-Log model.

### 5.4.2 Reliability, Validity, and Surprising Findings

The thing that struck us the most is that even out-of-the-box machine learning models didn't work at the beginning, and this was a surprise for us! Their negative R-squared made a strong and unexpected point about the very real applied reality of data science. This result supports a central thesis of this study, that the effect of a machine learning solution may be less related to the choice of a particular algorithm, but its end-to-end implementation pipeline. The fact that models didn't succeed is important: because it shows that the strength of these models is not natural, but behind a careful work of data preparation and optimisation.

Lastly, the small size of the final R-squared value for the best model (0.373 for the ‘CAKESTAND’) and very low positive ones for the other two products are, per se, an important ﬁnding. This result is a powerful, evidence-based observation that price and seasonality are crucial but certainly not the sole aspects shaping demand. This is the major limitation of the model that is fundamentally tied to the limitations of the data set i.e., it does NOT contain the important extrinsic variables like competitor prices, marketing spends, promotions, or the stock availability. This means that the model predictions are not reliable. There is still a significant amount of sales variance that the model has not captured-not because there is something wrong with the model, but because the data is not comprehensive.

This is a point, that for an SME is extremely important to consider. It means the frame, once again, should be used as directional model for forecasts as well as a potent “what if” scenario analysis tool not as a crystal ball predictor of specific sales numbers. This frank assessment of the model's shortcomings is key to using it responsibly in business and shows mature recognition of the gulf between statistics and the real world.

# 6. Conclusion and Future Work

This research was inspired by a marked and widening “analytics divide” in e-commerce. Although big corporates use big systems requiring many resources for dynamic pricing, the majority of the Small and Medium Enterprises (SMEs) cannot afford such systems and act upon their experience and gut feeling, while they do not have the necessary tools to keep up with the competition. This dissertation aimed to fill this gap by constructing, applying and validating a synthesis pragmatic hybrid framework that leverage the potential of new data science in a more effective and easily available way for business that do not have their own technical team. This concluding chapter summarises the main findings of this research offering an honest assessment of its limitations and proposes potential future research.

## 6.1 Conclusion

This project has successfully met its principal objective in designing and testing a hybrid analytical approach that meets an SME’s twin fold requirement: strategic, interpretative insight, and accurate, tactical forecasting. The course of our research changed course from its preliminary form to develop into a full comparative study where a classical Econometric Log-Log regression was up against two of the cutting-edge machine learning algorithms, Random Forests and XGBoost Regressors. The empirical findings offer convincing evidence that a dual-model approach is plausible for SME price optimisation. The main contributions and findings of this thesis can be summarised in three aspects:

* Econometrics Provides the Strategic 'Why'. The Log-Log regression model provides a useful method for obtaining a statistically significant and interpretable price sensitivity measure. However, even as a predictive aid its usefulness may be limited. But it can translate complex consumer behaviour models into one, theory-informed claim ("demand is very elastic") which is very useful.
* Machine Learning Provides the Tactical ‘What-If’. The literature review offers a clear judgment on the question, that it is the tuneable machine model that win the game of demand forecasting. It is observed that the Tuned Random Forest Regressor only the models which performs significantly better than others (explained 37.3% of sales variance of 'REGENCY CAKESTAND' product on unseen test data). That predictive dimension takes an SME out of merely telling the past to predict the future, so it can conduct tactical “what-if” scenario planning on what it is likely to sell at different price points a key power tool in the fast-changing world of e-commerce.
* The Process Outweighs the Algorithm. Arguably, the principle practical contribution of this dissertation is the concrete realization that the success of machine learning heavily relies on the strength of the implementation pipeline. The negative results obtained with the base-line ML methods were the first and in this last case crucial lessons learned on the fact that advanced algorithms are not "plug and play" systems. Their features revealed nothing obvious about dense or sparse sampled ingestions and secret intelligent engineering is required on both the feature processing and model parameters to realize their true predictive potential. This is a great lesson for SMEs, that the competitive edge is not in selecting a very complex algorithm, but in getting the data ready and the model tuned for the specific context of the business problem.

By addressing its fundamental research questions this dissertation has demonstrated a strong proof- of- concept for a Python driven hybrid framework that facilitates data-driven as-a-service pricing. It grapples with the trade-off between the interpretability of econometrics and the accuracy of machine learning to provide a powerful and accessible toolkit to help SMEs navigate the complexities of the modern digital economy.

## 6.2 Limitations

Although the goals of the project were reached, it is important to bear in mind the cons of it regarding the design and the data selection. Such limitations contextualize the findings of this study and indicate areas for future research.

There are some shortcomings in felicity and appropriateness issues in the dataset. An example of such a dataset is the Online Retail II dataset, which even if including real transactional data misses on a few features which are essential to build a full-blown pricing model.

* Lack of Competitor and Market Information: The models work in isolation from competition. Price increases and corresponding demand increases are dissected in a vacuum, and there is no room for external market momentum, such as a competitor promotion or general consumer exuberance.
* Missing Marketing and Promotion Data: No record of the retailer's own promotional activities is included in the dataset. A surge in sales in a particular week might have come from an email campaign or social media advertising, rather than a discounted price, and the existing models are not sophisticated enough to separate out such effects.
* Lack of Marginal Cost Data: A genuine profit optimisation model would need access to real COGS data. The current proposal is for a predictive model for demand and revenue, to use for forecasts, but with profit calculations based on cost assumptions.

The second group of limitations concerns the models. The Log Log model, also interpretable, however, did not correct for the tendency that price may be endogenous to inelasticity, which would result in biased elasticity coefficient estimates. And second, the “black box” character of the economically grounded Random Forest model is still a limitation in practice; the absence of a direct and handy measure of elasticity is why the superior Random Forest model needs to be combined with the econometric model.

Lastly the analysis was limited by necessity. The deep dive was for three products from one store. Although the detailed insights (like PED values or Random Forest performance compared to XGBoost) may be to some extent transferable to other product types or e-commerce models, they can in no way come claim generalizability to all.

## 6.3 Future Work

The model developed in this thesis provides a solid basis for number of interesting research directions. Following these extensions the proof concept will be turned into an industrial strength commercial price optimisation engine.

* Incorporating External Data: The easiest and highest-bang-for-your-buck step would be to combine sales data with other external variables. Potential future research can also include scraping web or using APIs to obtain competitor pricing information, promotional calendars or even macro-economic indicators. Their models would then include a much richer set of demand drivers, probably resulting in substantially more accurate predictions.
* Making It More Interpretable via Advanced IML: To overcome such a “black box” limitation of the Random Forest model, it is suggested that future research could make use of advance Interpretable Machine Learning (IML) tools. They could use something like SHAP (SHapley Additive exPlanations) to great effect next time. The SHAP values can break down individual predictions to see exactly how much the price, seasonality, and so on drove a specific sales forecast. This would reconcile the accuracy of the ML model with the business need of interpretability, getting the best of both worlds.
* Customer Segmentation and Price Discrimination: The current study treats customers as a homogeneous group. You may find it useful to first segment customers (if that's relevant, you can use Recency, Frequency, Monetary value (RFM) analysis) and to calculate price elasticities for each segment. It is also likely that frequent and valuable customers are linearly less price-sensitive than infrequent or low-value customers. Customized tiered pricing could yield significant incremental revenues.
* Building the Ultimate Optimisation Engine: A cool model would couple to live inventory and supply chain data. In the future, it would be interesting to develop a model that considers current inventory, the cost associated with holding inventory and the supplier lead time in its’ pricing suggestions. For instance, such a system could identify overstock and provide calculated recommendations on when and how to clear inventory by recommending a price drop to reduce carrying costs, now transitioning from price optimization to complete end to end profit optimization.

# References

Baker, W., Kiewell, D. and Winkler, G. (2020) 'Pricing in retail: Setting strategy for value creation', McKinsey & Company Retail Practice, pp. 1–12.

Bijmolt, T.H., Van Heerde, H.J. and Pieters, R.G. (2005) 'New empirical generalizations on the determinants of price elasticity', Journal of Marketing Research, 42(2), pp. 141–156.

Chen, D., Sain, S.L. and Guo, K. (2019) 'Data mining for the online retail industry: A case study of RFM model-based customer segmentation using data mining', Journal of Database Marketing & Customer Strategy Management, 19(3), pp. 197–208.

Cohen, M., Farias, V. and Thraves, C. (2017) 'Bias-variance tradeoffs in demand estimation and revenue optimization', Management Science, 63(10), pp. 3291-3308.

Ellickson, P.B. and Misra, S. (2008) 'Supermarket pricing strategies', Marketing Science, 27(5), pp. 811–828.

Ferreira, K., Lee, B. and Simchi-Levi, D. (2016) 'Analytics for an online retailer: Demand forecasting and price optimization', Manufacturing & Service Operations Management, 18(1), pp. 69-88.

Gallego, G. and van Ryzin, G. (1994) 'Optimal dynamic pricing of inventories with stochastic demand over finite horizons', Management Science, 40(8), pp. 999-1020.

Harrington, L.G. and Jenkins, M.A. (2018) 'Simplified price optimization for small and medium retailers: A practical framework', International Journal of Retail & Distribution Management, 46(11), pp. 1122–1138.

Kastius, K. and Schlosser, R. (2021) 'A reinforcement learning approach to dynamic pricing', Journal of Revenue and Pricing Management, 20(2), pp. 117-130.

Levy, M., Grewal, D., Kopalle, P.K. and Hess, J.D. (2004) 'Emerging trends in retail pricing practice: Implications for research', Journal of Retailing, 80(3), pp. xiii-xxi.

Marshall, A. (1890) Principles of Economics. London: Macmillan.

Misra, K., Schwartz, E.M. and Abernethy, J. (2019) 'Dynamic online pricing with incomplete information using multiarmed bandit experiments', Marketing Science, 38(2), pp. 226–252.

Montgomery, A.L. (1997) 'Creating micro-marketing pricing strategies using supermarket scanner data', Marketing Science, 16(4), pp. 315–337.

Tellis, G.J. (1988) 'The price elasticity of selective demand: A meta-analysis of econometric models of sales', Journal of Marketing Research, 25(4), pp. 331–341.

# APPENDIX

Appendix A: Python Code

* Onedrive: [u2734832\_DS7010\_final.py](https://uelac-my.sharepoint.com/:u:/g/personal/u2734832_uel_ac_uk/EZo3jI5GSr5LgT5Sfm5q3cIBKIQQfUWoIiXTJWImjwgWeQ)
* Github:<https://github.com/csgcode/PEDAnalysis/blob/main/u2734832_DS7010_final.py>

Appendix B: Dataset

* [Kaggle](https://www.kaggle.com/datasets/mashlyn/online-retail-ii-uci)
* [UCI](https://archive.ics.uci.edu/dataset/502/online+retail+ii)