NOVA IMS Information Management School

MDSAA

Master's Degree Program in

Data Science and Advanced Analytics

Taking advantage of Data Science practices to optimise Revenue Management Strategies in the Hotel Industry

Development of a Price Recommendation Model

Maria Inês Alves Ferreira da Silva

Master Thesis

presented as a partial requirement for obtaining a Master's Degree in Data Science and Advanced Analytics

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação
Universidade Nova de Lisboa

NOVA Information Management School

Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa

Taking advantage of Data Science practices to optimise Revenue Management strategies in the Hospitality Industry

Development of a Price Recommendation Model

by

Maria Inês Alves Ferreira da Silva

Master Thesis presented as a partial requirement for obtaining the Master's degree in Data Science and Advanced Analytics, with a specialization in Business Analytics.

Supervised by

Nuno António, PhD, NOVA IMS

December, 2024

STATEMENT OF INTEGRITY

I hereby declare having conducted this academic work with integrity. I confirm that I have not used plagiarism or any form of undue use of information or falsification of results along the process leading to its elaboration. I further declare that I have fully acknowledged the Rules of Conduct and Code of Honor from the NOVA Information Management School.

Maria Inês Alves Ferreira da Silva

Lisbon, December 02, 2024

ABSTRACT

With tourism continuing to grow every year, the hotel industry has become a key pillar of its success, generating significant revenue, with room bookings being one of the main contributors. Advances in technology have sparked greater interest in more sophisticated Revenue Management Systems. This dissertation focuses on developing a price recommendation model tailored to the unique needs of hotels in Portugal, aiming to predict optimal prices with accuracy. The dataset contained data from June 2018 until July 2021, containing 183812 observations and 22 variables. Before modelling, literature was conducted on pricing, forecasting and optimization algorithms for dynamic pricing tailored to the hotel industry. The data was tested using two approaches. In the first approach, we used normal regression models and in the second approach ensemble regression models. To ensure the model's accuracy, the quality of forecasts was measured using Negative Mean Squared Error (MSE). To further improve our best-performing models, we performed a hyperparameterization on them so we could see if we could improve our results. The results show that ensemble models, particularly tree-based methods, are effective in predicting dynamic pricing. By fine-tuning these models with techniques like Grid Search and Optuna, their performance was further enhanced, leading to more precise and reliable price predictions. The research also revealed that factors like booking patterns and competitor pricing are key in determining optimal prices. This dissertation brings a developed perspective on how data science can enhance revenue management strategies in the hotel industry. By creating a price recommendation model that uses machine learning techniques, it moves beyond traditional methods and focuses on dynamic, data-driven pricing decisions.

KEYWORDS

Pricing Optimization; Machine Learning; Revenue Maximisation; Regressive Models

Sustainable Development Goals (SDG):

TABLE OF CONTENTS

1. INTRODUCTION 1
1.1. Background and Motivation 2
1.2. Research Questions and Objectives 3
1.3. Thesis structure 3
2. LITERATURE REVIEW 5
2.1. Revenue Management Principles and Origins 5
2.1.1. Fundamental Concepts of Revenue Management 5
2.1.2. Historical Development 6
2.2. Revenue Management in the Hospitality Industry 7
2.2.1. Pricing 7
2.2.2. Forecasting 8
2.2.3. Optimization Algorithms for Dynamic Pricing 9
3. METHODOLOGY 10
3.1. Business Understanding 12
3.2. Data Understanding 13
3.3. Data Preparation 13
3.3.1. Data transformation 13
3.3.2. Missing values 14
3.3.3. Feature engineering 14
3.4. Modelling 15
3.4.1. Feature Selection 15
3.4.2. Models 17
3.4.3. Modelling results 21
3.5. Hyperparameterisation of the best models 21
3.5.1. Hyperparameterization Results Evaluation 24
4. RESULTS AND DISCUSSION 27
5. CONCLUSIONS AND FUTURE WORKS 29
BIBLIOGRAPHICAL REFERENCES 30
APPENDIX A – ORIGINAL DATASET VARIABLES 33

LIST OF FIGURES

Figure 1.1 - Distribution of Portuguese hotel revenue by month (2023) 2
Figure 3.1 - CRISP-DM Methodology (Chapman et al., 2000) 11
Figure 3.2 - Correlation Matrix Heatmap (Top Features) 17
Figure 3.3 - Normal Regression Models Accuracy 19
Figure 3.4 - Ensemble Regression Models Accuracy 20

LIST OF TABLES

Table 3.1 - Feature Engineer variables 15
Table 3.2 - Mutual Information values (Scores) of the top features 16
Table 3.3 - Normal Regression Models evaluation results 19
Table 3.4 - Ensemble Regression Models evaluation results 20
Table 3.5 - Overall Results 21
Table 3.6 - Grid Search hyperparameterisation parameters 22
Table 3.7 - Grid Search hyperparameterisation results 22
Table 3.8 - Optuna hyperparameterisation parameters 23
Table 3.9 - Optuna hyperparameterisation results 24
Table 3.10 - Hyperparameters for Best-Performing Models 25

LIST OF ABBREVIATIONS AND ACRONYMS

AI	Artificial Intelligence
AR	Augmented Reality
ET	Extra Trees
GBM	Gradient Boosting Machine
KNN	K-Nearest Neighbors
RF	Random Forest
RevPAR	Revenue per Available Room
VR	Virtual Reality

1. INTRODUCTION

Initially, Revenue Management depended mainly on manual procedures such as spreadsheets and rudimentary forecasting methods. When it came time to develop pricing strategies for their business, revenue managers made decisions based on historical data, trends, simple principles of supply and demand, and their judgement, often leading to poorly or not at-all-optimised pricing strategies (Kimes, 1989). This strategy, being room-centric, primarily impacted how prices were applied to rooms and how bookings were managed.

In the past years, Revenue Management has experienced critical changes that have changed how it is practised, primarily because of technological advancements. These emerging trends and innovations have shaped how industries, especially hotels, conduct business (Gatera, 2024).

Over time, the RMS has evolved and become more sophisticated, empowering revenue managers. It began to include a broader spectrum of complex data, such as historical data, data from competitors (comp sets), and more data associated with the market to which it belonged (such as data on events in the area or even meteorological data) (Ivanov & Zhechev, 2012). This evolution allowed revenue Managers to identify booking trends and make more complete and informed decisions, instilling confidence and control in their strategies.

These technologies have also developed over time, marking a turning point for RMS. The systems began to analyse more significant amounts of data (Big Data), which made them more complex. Machine learning models and predictive analysis were introduced to analyse and process this extensive data. This type of analysis has improved the accuracy of pricing decisions and made demand forecasting more precise (Anderson & Xie, 2010).

Forecasts and predictions in hotel pricing have undergone significant refinement alongside the evolution of Revenue Management Systems (RMS). Modern RMS now integrate advanced predictive analytics powered by various machine learning models (Kim et al.; S., 2020). These models include regression analysis for understanding relationships between variables, decision trees for segmenting customer preferences, and neural networks for complex pattern recognition in large datasets. They leverage vast datasets encompassing historical booking patterns, competitor pricing dynamics, market trends, and external variables such as local events and economic indicators (Zhang Y. et al., 2019). Machine learning algorithms enable hotels to forecast demand with unprecedented accuracy by continuously learning from new data inputs and adjusting parameters (Hassan A. et al., 2021). This capability enhances the precision of pricing decisions and optimises revenue strategies dynamically, providing a sense of reassurance and security in the face of a competitive hospitality landscape.

1.1. Background and Motivation

Tourism worldwide is one of the essential pillars in today's society, not only on an economic level (through job creation) but also on a social level, such as preserving a country's identity (in areas such as art, culture, and gastronomy) (Richards, 2018). According to the most recent United Nations World Tourism Organization analysis (UNWTO, 2023), international tourism finished 2023 at 88% of pre-pandemic levels, with an estimated 1.3 billion international arrivals.

Given the critical role of tourism in the global economy, it is essential to recognize the hotel industry as one of its foundational pillars. As a primary service provider for international travellers, the hotel industry generates substantial revenue and supports a wide range of related businesses (Sampaio et al., 2024).

According to the *Preliminary overview of 2023* from Turismo de Portugal and INE (Statistics Portugal), the tourism sector in Portugal exceeded everyone's expectations, surpassing pre-pandemic levels and making up for 16% of its GDP (Horwath HTL Hotel, 2023). In 2023, 30 million tourists visited Portugal, accounting for 77.2 million hotel stays. Figure 1.1 displays the monthly distribution of hotel revenue that hotel stays generated in Portugal, highlighting the peak occupancy and revenue periods. We retrieved this information from *TravelBI* by Turismo de Portugal, which provides the latest insights on tourist activity in the country.

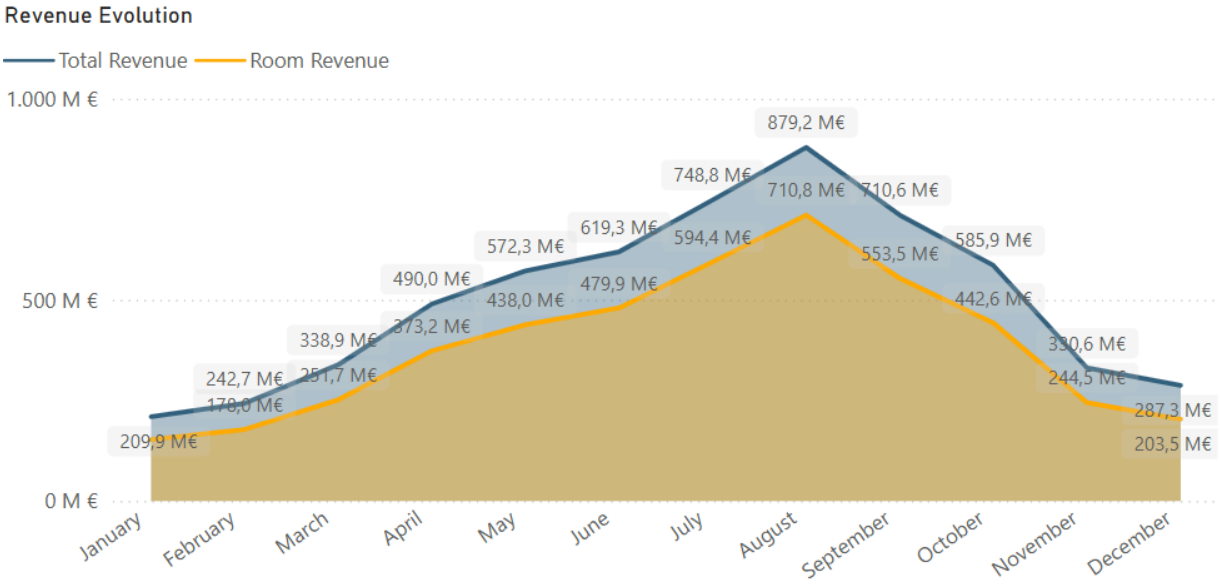

Figure 1.1 - Distribution of Portuguese hotel revenue by month (2023)

This sector relies on multiple revenue sources, including room bookings, food and beverage sales, and event hosting, all of which are highly sensitive to price and demand fluctuations. In this context, optimizing room pricing has become increasingly essential, as competition and market dynamics require hotels to adapt quickly to changing conditions (Kimes, 2010). Leveraging data science and machine learning techniques, hotels can now fine-tune pricing strategies based on real-time demand forecasts and evolving market trends, leading to

maximized revenue, enhanced operational efficiency, and a more tailored guest experience (Egger, 2022).

Machine learning (ML) forecast models emerge as a critical tool for anticipating future trends and optimising revenue management strategies (Ivanov & Zhechev, 2012). Unlike traditional methods, ML models handle complex data, encompassing booking patterns, guest preferences, pricing dynamics, and market fluctuations. By taking advantage of advanced algorithms, these models enhance forecast accuracy and precision, providing hoteliers with actionable insights into demand forecasts, occupancy rates, and revenue potentials with unprecedented granularity (Henriques & Pereira, 2023).

Given the economic importance of tourism and the hotel industry's reliance on effective pricing strategies to drive revenue, this thesis aims to explore a more data-driven approach to pricing strategies. Traditional methods often struggle to keep up with the complex and ever-changing factors affecting demand. Machine learning and data science offer promising ways to handle this complexity, and that's exactly where this research steps in. The goal is to develop a price recommendation model that empowers hotels to make smarter and more efficient pricing decisions. By introducing this model, this thesis aims to support hotels in maximizing revenue and gaining a stronger competitive edge in the tourism market.

1.2. Research Questions and Objectives

This study aims to create a data-driven pricing recommendation model specifically tailored to the needs of Portuguese hotels. Given this, our research objective is to develop a model to identify and predict future pricing trends based on historical pricing data and booking patterns. This approach will enable us to accurately forecast pricing demand while considering various influencing variables (seasonality, competition, customer demand, …).

By analysing our model's performance, we aim to contribute new insights to future research and implement innovative data-driven approaches in the hotel industry. During our process, we hope to answer the following investigation questions.

1. Which variables or combinations of features have more influence on pricing predictions?
2. Which machine learning model offers the best balance between predictive accuracy and practical implementation, and how can its performance be optimized further?

1.3. Thesis structure

The thesis will be structured as follows. Chapter 1, Introduction, introduces the topic that will be further studied and reviewed in the following chapter. Chapter 2, Literature Review, presents a more in-depth look into the research in this thesis. In this chapter, we position the research objectives in the space of previous research on the topic, highlighting key findings (such as strengths and weaknesses of existing research) and identifying gaps in the literature to which our research can contribute.

Chapter 3, Methodology, gives a general introduction and overview of the problem and how it is proposed to be solved. In this chapter, we describe the methodology used in the study, including the research design, data collection and analysis techniques, and ethical considerations.

Chapter 4, Results and Discussion, presents the study's results, including data analysis and interpretation, and compares our results with existing research.

Finally, Chapter 5, Conclusion, summarises the research, its implications for the field, key results, and how the research questions were answered. In addition, this chapter contains a summary of the limitations encountered throughout the research and recommendations for future research.

2. LITERATURE REVIEW

This section examines and compares the relevant literature on the topic. Each section has subtopics to address studies and approaches to complete our analysis.

2.1. Revenue Management Principles and Origins

In this initial chapter, we will establish foundational concepts, origins, and relevance of Revenue Management in the hostel industry.

2.1.1. Fundamental Concepts of Revenue Management

Revenue Management is crucial in various industries, such as the airline and hospitality sectors. Kimes and Wirtz (2003) define revenue management as applying information systems and pricing strategies to allocate the suitable capacity to the right customer at the right price and time. This ultimate objective is achieved through demand-management decisions, which require knowledge of a wide range of demand-management characteristics (Talluri & Van Ryzin, 2005).

The airline industry first developed the concept in the 1970s, and it remains a crucial practice for optimising profits through demand-based pricing and capacity control. Industries such as hotels, restaurants, golf courses, car rentals, and other companies - as different as they may be - have one thing in common: they all practice revenue management (Kimes, 1989).

Initially, many industries adopted revenue management to tackle specific business challenges. Over time, however, companies discovered that this approach wasn't just a problem-solver—it became a powerful tool for boosting profits and even sparked technological advances across various areas of their operations (Cross, 1997; Weatherford & Bodily, 1992). Managers found that revenue management brought unexpected benefits, like greater flexibility in responding to market changes and valuable insights into customer behaviour, allowing for smarter pricing decisions and closer alignment with customer needs (Talluri & Van Ryzin, 2004; Phillips, 2005). This shift made companies more agile and attuned to both customer preferences and market demands. Ultimately, the strength of revenue management lies in its ability to maximise returns from fixed assets, as the marginal benefit of each service provided is naturally limited by physical capacity (Hanks, Noland & Cross, 1992; Kimes, 1989).

Some conditions must be met for Revenue Management techniques to be applied in a business based on various economic fundamentals and assumptions (Kimes, 2003; R. L. Phillips, 2005). The conditions are as follows:

- Capacity is limited and immediately perishable.
- Customers book capacity ahead of time.
- Opening and closing predefined booking classes change prices.

Of course, while certain revenue management principles can be applied across various industries, each industry adapts this practice to its own business, considering their specific characteristics. What is successful for some might not be for others.

2.1.2. Historical Development

In 1978, the US Airline Deregulation Act, where the U.S. Civil Aviation Board (CAB) loosened control of airline prices and, from that moment on, existing airline companies could now alter their prices, timetables, and offerings without needing approval from CAB (Talluri & Van Ryzin, 2005). As the level of competition among airlines rose, they tried to operate their planes as efficiently as possible.

Due to these changes, the scientific community was prompted to devise a new management strategy known as Yield Management, a system where the company used forecasting methods (from operations research and applied mathematics) to predict demand and adjust prices accordingly (McGill & Van Ryzin, 1999). Yield Management was created to fulfil the need to occupy a minimum number of plane seats to cover fixed operating expenses (Belobaba, 1987). Once these fixed costs were covered, the remaining capacity could be sold at higher rates to maximize revenue (Weatherford & Bodily, 1992). It quickly became a critical tool for airlines to maintain profitability in a highly competitive industry. Airlines optimized their revenues by dynamically posting different fares and controlling the seats available at different fare levels (Smith, Leimkuhler, & Darrow, 1992).

After successfully implementing and improving this new practice in the airline industry, Revenue Management was born. Both practices aim to increase revenues and maximise profits, but they are two different concepts (Liberato et al., 2023). With this being said, Revenue Management expanded in the mid-1980s and early 1990s into other areas, primarily in the hospitality industry. Technological systems were introduced very early to speed up data analysis and price setting for revenue optimisation (Cross, 2016).

With the technological advancements in the 1990s (such as the availability of personal computers and the development of more sophisticated software), Revenue Management became more accessible to smaller hotels and independent properties.

Since then, Revenue management has evolved to be a common practice in a wide range of industries, and the practice itself has faced several types of technological improvements, primarily relying on optimisation and forecasting algorithms (Ivanov, 2014). Integrating technology into hotel processes has brought innovation and opened a range of opportunities and advantages that are constantly changing. According to the Lodging Technology study Embracing Mobility & Self-Service (Hospitality Technology, 2023b), 73% of hoteliers consider implementing technology in their processes essential to a hotel's performance. These technologies include Artificial Intelligence (AI), Augmented Reality (AR) and Virtual Reality (VR) solutions.

2.2. Revenue Management in the Hospitality Industry

In the first chapter of the literature review, we introduced the concept of revenue management in two ways: through a brief historical introduction and by discussing its essential principles. Since our study is based on developing a predictive pricing model, in this chapter, we will look at the most critical literature that could pave the way for our methodology.

2.2.1. Pricing

Effective revenue management pricing strategies are crucial for maximising profitability in the hotel industry. For this to happen, the revenue manager or team is tasked with aligning the company's pricing strategies with the informed customers' willingness to pay. By achieving this alignment, customers recognise value and fairness in the prices, leading to maximised revenue and profitability for the business (Hayes & Miller, 2011).

While some industries use price-based RM, others find quantity-based RM more appropriate. Hotels use elements of quantity management, such as setting minimum stay requirements and managing room inventory. Still, their primary focus remains on price-based RM practices, such as adjusting prices to align with demand and maximise revenue. Hotels use various pricing policies to structure these prices, trying to protect the revenue they collect from periods with higher demand when hotels raise rates and periods of lower demand when lower rates make sense and extract as much value as possible for the rooms (Hayes & Miller, 2011).

Price-based RM in hotels involves adjusting room rates based on various factors, whether internal (revenue goals or service offerings) or external (market demand, competition, consumer perceptions, and preferences) (Talluri & van Ryzin, 2004; Abrate & Viglia, 2016). This strategy, often called dynamic pricing, allows hotels to charge higher rates during periods of high demand and lower rates during periods of low demand, thus maximizing revenue potential (Enz & Canina, 2012).

Dynamic pricing is a strategy whereby businesses set flexible product or service prices based on market demands (Elmaghraby & Keskinocak, 2003). In more familiar terms, dynamic pricing is a flexible pricing mechanism made possible by recent technological advances. As opposed to traditional static pricing, it's not adjusted based on the cost of production. Dynamic pricing gives businesses more flexibility to price goods or services based on market demands and is frequently used in hospitality, travel, entertainment, retailing, and electric utility services (McAfee & te Velde, 2006). When choosing dynamic pricing in markets, key factors include consumer price sensitivity, production and administrative costs, variable demand, the degree of operating leverage, and various other pricing strategies such as mixed pricing, portfolio effects, and revenue management incentives (Talluri & Van Ryzin, 2004; Phillips, 2005). Supply and demand are the primary drivers of dynamic pricing (Gallego & Van Ryzin, 1994).

Abrate et al., 2019 studied the impact of price variation (or dynamic pricing) over time on hotel revenue maximisation. They started by explaining some constituents in the effects of dynamic

price variation on revenue maximisation, which were these four domains: intertemporal price discrimination, which suggests that businesses can charge different prices over time to capitalise on variations in consumer willingness to pay; Price fairness, which investigates how variations in prices affect consumers' perceptions of fairness; Inventory controls, that focuses on managing inventory effectively in the context of dynamic pricing; and lastly, organisational culture, that explores organisational factors that influence the adoption and implementation of dynamic pricing strategies. Their proposition was a hedonic revenue model, where, based on some aspects, they would test the relationship between dynamic price variability and revenues. Their results suggest that hotels could apply more dynamic price variability strategies and increase the period on the variability of prices because it would directly impact revenue maximisation.

This highlights the importance of understanding the multifaceted nature of pricing decisions in the hotel industry. Unlike simplified assumptions, pricing decisions are dynamic and complex, varying across different periods. Numerous factors contribute to these variations. Variables such as location within a city, target market selection, and technical aspects, including hotel age, room count, and food and beverage revenue, all play significant roles. Moreover, prices fluctuate concurrently due to differences in service offerings, comfort levels, and seasonal demand within the same hotel, reflecting distinct pricing tiers perceived by guests. This multifaceted nature of pricing decisions underscores the need for a comprehensive approach to revenue management in the hotel industry (Hernández et al., 2021).

2.2.2. Forecasting

In the early days of forecasting in revenue management for hotels, the focus was on occupancy and average rate forecasts. However, over the past decade, the role of forecasting has evolved significantly, becoming a powerful tool for decision-making. The ability to predict future demand and plan capacity usage accordingly is now essential in revenue management, particularly when demand and capacity are complementary, and capacity cannot be easily adjusted (Webb et al., 2020).

Historically, forecasting models fall into three main categories: Historical, Advanced Booking, and Combined models (Weatherford & Kimes, 2003). Historical models, which use past data and traditional techniques like linear regression, provide a strong foundation for predicting demand. Advanced Booking models focus on reservation patterns within specific periods to predict future demand, while Combined models merge historical and booking data to enhance forecasting accuracy.

Despite the success of these models in many sectors of tourism and hospitality, they still struggle with predicting demand driven by more complex factors. One promising advancement in this area is the use of Neural Networks. These models, inspired by the human brain, learn from data and adjust accordingly, offering improved modelling capabilities to handle complex, large datasets (Montesinos et al., 2022). Neural networks have recently been applied to

forecasting in the hotel industry, a field where they were previously underexplored. For example, Lee et al. (2020) demonstrated that combined neural network models could enhance forecasting accuracy for hotel demand in the short to medium term.

Although neural networks have shown great promise, the literature comparing their performance with traditional statistical methods, like ARIMA, remains limited. Studies by Adebiyi et al. (2014) and Claveria and Torra (2014) have shown that ARIMA models sometimes outperform neural networks, but other research, including that by Zhang et al. (1998), highlights the superior forecasting ability of neural networks. This gap in the literature presents challenges for hotel managers seeking to adopt the best practices and most effective technologies. Without more robust studies on forecasting in the hotel industry, there is a risk that technological advancements, especially in machine learning, will not be fully leveraged. Collaborative research between academia and industry could help bridge this gap and lead to more effective forecasting models that drive better revenue management practices (Xie et al., 2018).

2.2.3. Optimization Algorithms for Dynamic Pricing

Dynamic pricing is about flexibility—adjusting prices based on real-time market demand. In the hotel industry, where guest demand can change daily or even hourly, this approach is essential for maximizing revenue and staying competitive (Phillips, 2005). Optimization algorithms play a crucial role in dynamic pricing, enabling hotels to adjust prices based on factors like forecasted demand, current occupancy, and competitor rates. This adaptability empowers hotels to make data-driven pricing decisions, responding to short-term demand shifts and broader revenue goals (Bodea & Ferguson, 2014).

To manage dynamic pricing effectively, hotels often rely on optimization algorithms like linear programming, mixed-integer programming (MIP), and heuristic methods. Linear programming helps hotels find the optimal price while respecting constraints like room availability (Talluri & Van Ryzin, 2004). MIP offers more flexibility by handling both continuous and discrete variables, useful for adjusting room rates by type, booking period, and seasonal demand (Ivanov, 2014). For more advanced pricing, techniques like reinforcement learning and stochastic programming allow algorithms to learn and adjust based on guest responses and demand uncertainty (Zakhary et al., 2011; Bodea & Ferguson, 2014).

Fine-tuning these models is crucial to ensure they capture complex pricing patterns. Tools like Grid Search, which exhaustively evaluates specified hyperparameter combinations, can significantly improve model performance by systematically selecting the best configuration for predictive accuracy (Pedregosa et al., 2011). This approach helps enhance the effectiveness of the price recommendation model for hotels by better-capturing demand fluctuations and optimizing revenue outcomes.

However, to ensure these models' reliability, benchmarking their performance is essential. Comparing different machine learning models—such as decision trees, random forests, and

gradient boosting—using metrics like mean squared error (MSE) and R-squared values, helps assess their accuracy (Henriques & Pereira, 2023). Cross-validation further improves reliability by ensuring the models generalize well to unseen data (Kimes & Wirtz, 2003). By establishing these performance benchmarks, this research aims to guide hoteliers in selecting the most suitable models to optimize pricing strategies.

3. METHODOLOGY

In the previous chapter, we introduced the concept of Revenue Management and delved into its practices. Ultimately, we concluded that the path that would make the most sense would be to build a predictive price model for hotels. As we will use data-driven methods (specifically,

machine learning models) to organise our methodology, we chose to use the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology (Chapman et al., 2000). The method will be applied throughout our methodology and project application. It follows a data-driven orientation with 6 phases, as shown in Figure 3.

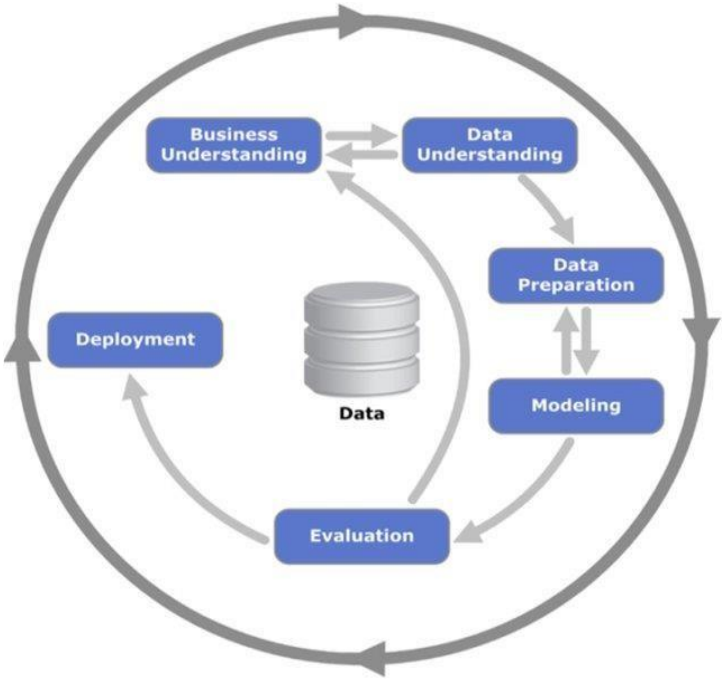

Figure 3.1 - CRISP-DM Methodology (Chapman et al., 2000)

If we look at the image above, we can see that we can revisit previously carried out steps based on the results and feedback we get in other phases to improve our model. The natural cycle of a project that involves building a predictive model presupposes a loop between the modelling and evaluation phases, as different models will give different results.

The six phases that form part of this process are as follows:

- Business Understanding – This initial phase includes project objective, understanding business goals, and assessing the situation from a data perspective;
- Data Understanding – At this stage, we collect and explore the available data to gain insights into its quality, structure, and potential relevance to the project.
- Data Preparation – It focuses on cleaning, transforming, and integrating the data to prepare it for analysis;
- Modelling – After understanding and processing the data, we select and apply the appropriate modelling techniques to build predictive models at this stage. It includes tasks such as modelling techniques, training and evaluating models using training data, and iteratively refining models to improve performance;
- Evaluation – In the evaluation phase, we assess the performance of the models developed in the previous phase, comparing different models to select the best-performing one;

- Deployment – The final phase focuses on deploying the selected model into operational use and monitoring its performance over time.

Throughout this chapter, we will answer the research questions posed in the introduction to this thesis through the steps proposed in the methodology applied. This methodology was chosen because it provides a proven and practical framework for data science projects. Its structured, flexible, and collaborative approach helps to reduce risk, ensure alignment with business goals, and maximise value.

3.1. Business Understanding

In the first stage of the CRISP-DM methodology, Business Understanding, we start by defining the business objectives of our price recommendation model within the context of the hotel industry. We will focus on understanding the hotels in the dataset business environment, with a specific focus on its pricing strategies over recent years, and how they can be optimised. This includes a review of the hotel's historical room pricing and occupancy data from the past few seasons, considering factors such as seasonal demand that influence pricing decisions. The hotel employs various pricing strategies, including dynamic pricing, Room Categories, Discounted Rates, and Special pricing for corporate and group reservations.

We first outline a detailed project plan, including phases such as data collection, preparation, modelling, evaluation, and deployment. A timeline with specific milestones ensures that we stay on track, and resource allocation plans detail the personnel, tools, and technologies required to complete the project successfully.

We then determine the business requirements, establishing KPIs like RevPAR, ADR, and occupancy rates to measure success. We acknowledge constraints such as budget limitations and technological capabilities. Our desired outcome is to develop a data-driven model that provides accurate and reliable price recommendations, leading to improved revenue and competitive advantage. To achieve this, we formulate data mining goals by converting our business objectives into a clear problem statement: 'Develop a model to predict optimal room prices based on historical booking data and external factors.' The model's accuracy will measure success.

By thoroughly addressing these elements in the business understanding step, we set a solid foundation for the next stages of the CRISP-DM methodology to achieve the objective of finding the best set of prices which maximise revenue.

Lastly, it is important to mention that this work does not aim to assess whether adding or removing specific hotel amenities or services directly correlates with profitability, considering their associated costs. While certain factors, such as service quality and customer experience, undoubtedly influence the hotel’s profitability, this model seeks to enhance pricing strategies without delving into operational costs or capacity-related decisions. The ultimate goal is to

improve revenue generation through data-driven pricing recommendations, benefiting the hotel's overall financial health and long-term sustainability.

3.2. Data Understanding

The company XLR8 Revenue Management System, which specialises in developing and implementing AI-based solutions to maximise the day-to-day tasks of a hotel revenue manager, provided a dataset required to build the price recommendation model proposed in this thesis. The dataset comprises data from two sources: PMS data and competitive set data. Anonymity is necessary for the hotels in question in this dataset, so none of the entries contain their names. Given this, we received an Excel file with data from June 2018 until July 2021, containing 183812 observations and 22 variables, as presented in Appendix A.

The entire practical component of the work was conducted using Python and Jupyter Notebook. Upon importing the file into Jupyter Notebook, we were able to analyse the data and identify the necessary steps for the data preparation stage.

3.3. Data Preparation

As we mentioned at the beginning of this chapter, in this phase, we carry out the necessary transformations to the initial dataset to prepare it for the subsequent phases of analysis and modelling; this includes cleaning the dataset by converting 'object' columns to 'numerical', check for duplicated values (which did not exist), removing rows that failed to convert, checking for missing and null values and dropping them to ensure we apply the models to credible data and obtain reliable results.

3.3.1. Data transformation

After importing and analysing the data, it was essential to perform some data transformation steps so that our data would be ready to be used in our models. We first converted object columns to numeric columns. Object columns often contain categorical data, which cannot be directly used in algorithms that require numerical input (which is our case). By transforming these columns into numerical format, we enable a broader range of techniques and improve the performance of our models. We identified the object columns in the dataset, which were 'arr_pickup_leadtime', 'arr_pickup_target', and 'arr_target'. Initially, these columns were numerical, but since they had some error values (#NAME), they were considered objects by Pandas. We converted them into a numeric format. This step ensures the data is clean and usable for machine learning, allowing the model to understand better how these features influence pricing and improve its predictions. In this step, the records that cannot be converted to numeric format are dropped. After this step, we checked for duplicate values and found none.

After some analysis, we concluded that some values did not make sense, and removing the rows containing them would be beneficial. By eliminating these values, we update the dataset

to focus only on the most relevant features, which improves model performance and generalisation to new data. Since our goal is to build a price recommendation model, with the feature ‘price’ as our target variable, we started by dropping all the rows where the ‘price’=0. We noticed that when the column ‘is_restrict’=1, the price is always 0, so we drop the rows when ‘is_restrict’=1. It also made sense to drop the rows when "sold_out=1" because the price was always 530 since it does not make sense to have price values when the hotel is sold out (we assume that the value 530 is a default value). So, we drop rows with sold_out=1.

3.3.2. Missing values

The dataset had missing values in some features. We were missing 104 values in most columns (including in the ‘price’ column), so we removed them.

Then, we had many missing values in the columns ‘arr_pickup_leadtime’ and ‘arr_target’. We had two possible solutions: we removed all the rows with missing values or performed an imputation using the KNNimputer. Imputing using the ‘KNNImputer()’ involves filling in missing values in a dataset using the k-nearest neighbours (KNN) algorithm. This method uses the similarity between data points to estimate the missing values. We chose to use the KNNImputer method because removing rows with missing values results in the loss of potentially valuable information from other features that are complete. Imputation allows us to preserve these rows and use the information from different features to estimate missing values.

3.3.3. Feature engineering

Feature engineering is a crucial step in developing predictive models, particularly in the context of price recommendation models. So, before training the models, we created new variables from the original dataset to enhance the model's performance by providing more informative features. We created the following variables:

The Competitive Price Difference feature captures the difference between the hotel price and the median price of the competitive set. This variable directly measures how competitively the hotel prices its rooms compared to its competitors. This way, the model can better assess the hotel’s pricing strategy relative to the market, which can be crucial for future revenue management strategy optimisation.

The Competitive Set Occupancy rate is estimated as the ratio of rooms sold to the total number of rooms in the competitive set. This variable indicates the overall demand in the competitive market. High occupancy rates in the competitive set can signal strong market demand, influencing pricing decisions. By incorporating this feature, the model gains insight into market trends and can make more informed pricing recommendations.

Booking pace is calculated as the number of rooms booked per day, defined as the number of rooms sold divided by the lead time. This variable is crucial as it indicates the rate at which bookings are made. A higher booking pace may suggest high demand or successful marketing efforts, prompting adjustments in pricing strategies. This variable allows the model to adapt to real-time booking trends and optimise pricing accordingly.

RevPAR is a standard industry metric calculated as the revenue earned from room sales divided by the number of available rooms. This variable is a crucial indicator combining occupancy and average rate performance. Including RevPAR in the dataset allows the model to understand the hotel's overall revenue efficiency, providing a holistic view of revenue performance that can guide pricing decisions.

Lastly, ADR is calculated as the total room revenue divided by the number of rooms sold, representing the average revenue earned per occupied room. This variable is important because it directly reflects the hotel's pricing power. By incorporating ADR, the model can assess how well the hotel captures revenue per booking, which is vital for setting optimal room rates.

By including these new variables, we enhance the model's ability to capture essential aspects of the competitive market, booking behaviour, and revenue performance. The resulting variables are shown in Table 3.1.

Table 3.1 - Feature Engineer variables

Column	Type	Description	Sample
comp_price_diff	Float	The difference between the hotel's price and the median price of the competitor set	-15.5
comp_occupancy_rate	Float	Estimated as the ratio of rooms sold to the total number of rooms in the competitive set	0.85
booking_pace	Float	The number of rooms booked per day, calculated as the number of rooms sold divided by the lead time	2.35
RevPAR	Float	Calculated as rooms_rev/rooms_available	123.45
ADR	Float	Calculated as room_rev/rooms, representing the average revenue earned per occupied room	145.67

3.4. Modelling

3.4.1. Feature Selection

Before applying models to our data, we need to find the features that impact the variable we predict most. We do this step before initial modelling to ensure that the model is trained only with the most relevant variables, reducing dimensionality and preventing overfitting. We used the SelectKBest method to calculate the mutual information values (scores) between each feature and the predicted variable. This method selects the top k features based on a scoring function. The mutual information between two variables is 0 if these variables are entirely

independent, and the higher the value, the higher the dependency. This means that using the features with a higher score results in better results for the models. We present the scores obtained in Table 3.1. We chose to have the score cutoff at 1.3. We also decided to include the new features based on the initial comp set features in the initial modelling. Incorporating comp set features helps make the price recommendation model more robust, and competitive, allowing for more accurate demand forecasting and price sensitivity analysis. This leaves us with the features: "rooms_rev_target", "arr_target", "RevPAR", "ADR", "arr_pickup_target", "price_pickup", "price_pickup_target", "rooms_rev", "week_number".

Table 3.2 - Mutual Information values (Scores) of the top features

Feature	Score
week_number	1.32859
rooms_rev	1.37494
price_pickup_target	1.58286
price_pickup	1.58291
ADR	1.61788
arr_pickup_target	1.6206
RevPAR	1.85305
arr_target	1.99306
rooms_rev_target	2.0009

We then examine the heatmap of the correlation between these features shown in Figure 3.2. We find that the price_pickup and price_pickup_target have a correlation value of 1. If we drop only one of these features, we lose no information. We also observe that rooms_rev_target correlates highly (0.83) with price_pickup and price_pickup_target. Removing two of these features' results in a negligible loss of information for the models. Since the rooms_rev_target feature has the highest score, removing both price_pickup and price_pickup_target makes sense.

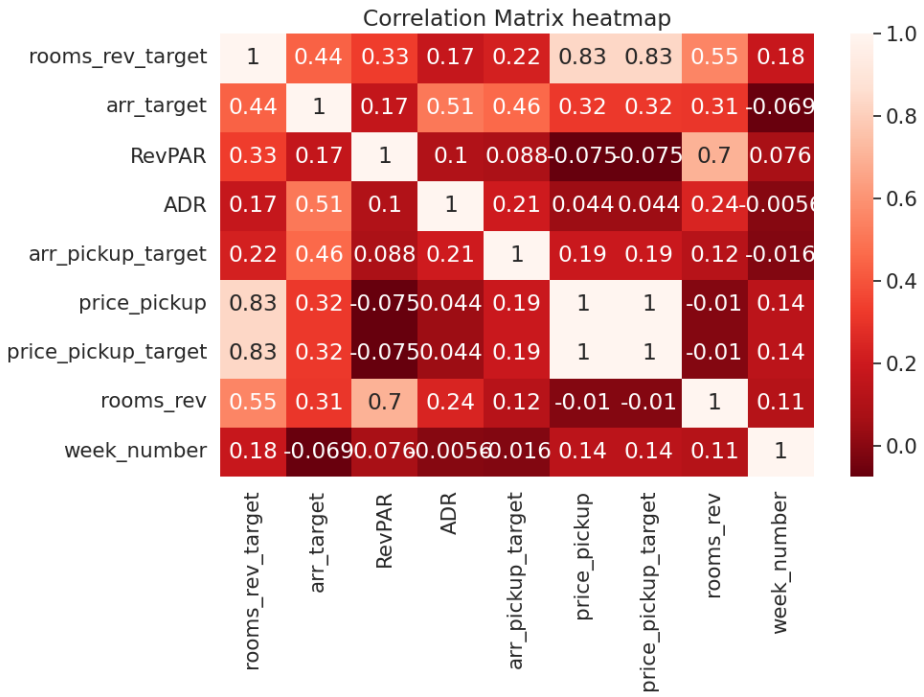

Figure 3.2 - Correlation Matrix Heatmap (Top Features)

3.4.2. Models

We split the dataset into three parts: training, validation, and test sets, using a 60-10-30 ratio. 60% of the data was assigned to the training set, 10% to the validation set, and 30% to the test set. The training set is used to teach the models by letting them learn from historical data, helping them make predictions for the future. The validation set serves as a checkpoint to assess how well the model is performing on unseen data, allowing us to fine-tune the model's hyperparameters and select the best-performing models based on prediction quality. Finally, the test set is used to compare the performance of the different models and determine which one delivers the most accurate predictions.

To accomplish this, we used the 'train_test_split' function from the Scikit-learn library in two stages. First, the data was split into the training set and a temporary set (which would be further split into validation and testing). Then, the temporary set was divided into the validation and test sets using the appropriate proportions. A random seed of 5 was specified to ensure reproducibility. After splitting the data, feature scaling was applied to standardize the input variables. The 'StandardScaler' was utilized for this purpose, where the 'fit_transform' method was applied to the training set to compute the mean and standard deviation. The 'transform' method was then used on both the validation and test sets to ensure consistent scaling across all data subsets.

In this modelling stage, we tested a few regression models since these are the most frequently used for predictive analysis (Manasa et al., 2020). The choice of models was guided by the need to balance interpretability, computational efficiency, and predictive performance. We divided the modelling into two approaches. The first approach used Normal regression models

to establish baseline performance and provide interpretable results. The second approach uses Ensemble regression methods to enhance predictive accuracy and robustness, leveraging their ability to capture complex patterns and relationships in the data.

Before modelling, it was important to define the error measure used to assess the quality of the forecasts. For this, the Negative Mean Squared Error was chosen. The reason behind this decision is that negative MSE penalizes larger errors more significantly, making it a useful metric when trying to minimize substantial deviations between predicted and actual values. This is particularly relevant in revenue management, where large prediction errors can have a more profound impact on pricing and revenue strategies. MSE provides a clear indication of how well the model is performing by measuring the average squared difference between predicted and actual values. In practice, it means that, on average, the model's predictions are off by a certain amount in terms of squared error. This allows us to evaluate the model's performance, with lower MSE values signifying better accuracy in forecasting.

In our first approach to building the model, we tested several regression algorithms to see how well they could predict hotel prices. The models we chose—Logistic Regression, K-Nearest Neighbors (KNN), Linear Support Vector Regression (LinearSVR), Decision Tree Regressor, and Linear Regression—were selected to cover a range of approaches. This included simple and easy-to-interpret models like Logistic and Linear Regression, as well as more flexible ones like KNN, which looks at similar data points, and Decision Trees, which handle complex patterns. By trying these different methods, we aimed to find the best fit for capturing the relationships in the data and making accurate price predictions.

To evaluate the performance of each model, we implemented a k-fold validation approach using 10 folds. This process involved splitting the training data into 10 equal-sized folds. For each fold, the model was trained on 9 folds and tested on the remaining fold, ensuring that every data point was used for both training and testing. We used the ‘cross_val_score’ function from Scikit-learn with the Negative MSE as the scoring metric, allowing us to measure the regression accuracy of each model and the quality of the forecasts. After evaluating each model, we can observe in Figure 3.3 that the K-nearest neighbours and Decision Tree had the best MSE values.

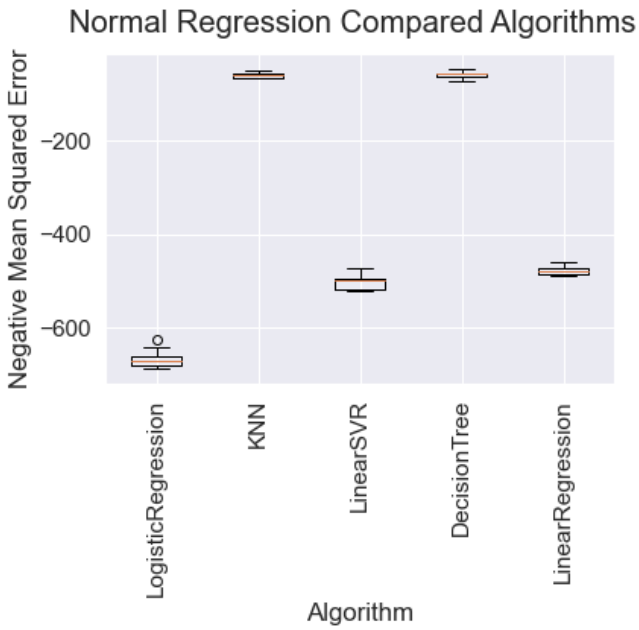

Figure 3.3 - Normal Regression Models Accuracy

Each model was then fitted on the training data and evaluated on the test data using three key performance metrics: Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-squared (R2). These metrics provide insights into the prediction accuracy and the model's ability to explain the variability in the data. The evaluation results revealed varying performance across the models, as shown in Table 3.3.

Table 3.3 - Normal Regression Models evaluation results

Model	Mean Squared Error (MSE)	Mean Absolute Error (MAE)	R-squared (R2)
Logistic Regression	656.4	16.13	0.4
K-Nearest Neighbors	48.35	2.98	0.96
Linear SVC	500.97	16.46	0.54
Decision Tree	49.36	2.43	0.95
Linear Regression	474.27	16.82	0.56

The KNN Regressor and Decision Tree Regressor significantly outperformed the other models, achieving the lowest MSE and MAE values and the highest R-squared scores (0.96 and 0.95, respectively).

In our second approach, we chose AdaBoost, Gradient Boosting, Random Forest, Extra Trees, and LightGBM because they are great at handling complex relationships in data and tend to be more accurate and reliable by combining multiple predictions. These models are especially well-suited for tasks like dynamic pricing, where capturing patterns and making precise predictions are crucial. To evaluate the performance of these models, a cross-validation technique using KFold with 10 splits was adopted, with each fold serving as a validation set while the remaining folds formed the training set. This technique ensures that each model's performance is assessed across multiple subsets of data, providing a more unbiased estimate. As we did in the first approach, we used the Negative Mean Squared Error (MSE) as the scoring

metric to measure the average of the squared differences between the predicted and actual values, with lower values indicating better performance. After evaluating each model, we observed that the Random Forest and Extra Trees had the best MSE values, as shown in Figure 3.4.

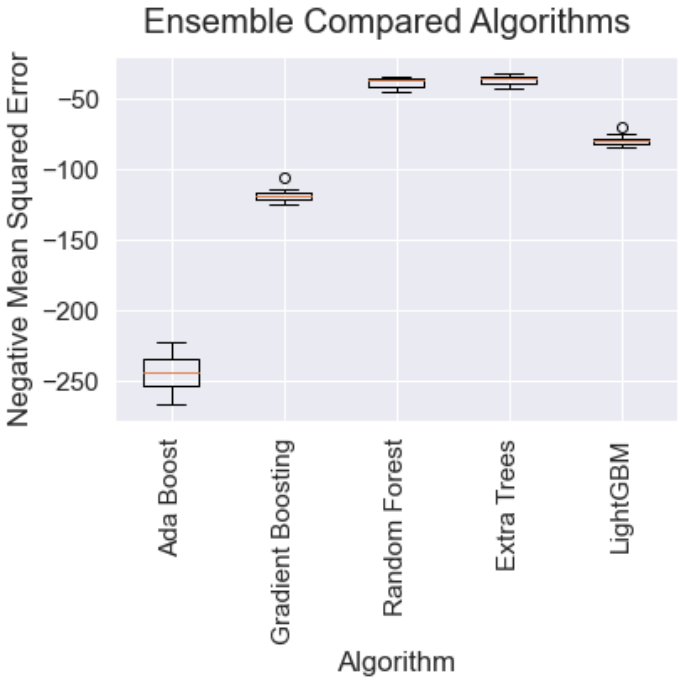

Figure 3.4 - Ensemble Regression Models Accuracy

After each ensemble model was instantiated and trained on the training set, predictions were made on the test set, and performance metrics such as Mean Squared Error (MSE), Mean Absolute Error (MAE), and R-squared (R2) were computed to evaluate their predictive accuracy. These metrics provide insights into how well each model performs with unseen data. The evaluation results revealed varying performance across the models, as shown in Table 3.4.

Table 3.4 - Ensemble Regression Models evaluation results

Model	Mean Squared Error (MSE)	Mean Absolute Error (MAE)	R-squared (R2)
AdaBoost	235.03	10.73	0.78
Gradient Boosting	116.07	7.05	0.89
Random Forest	32.91	2.36	0.97
Extra Trees	32.96	2.22	0.97
Light GBM	73.94	5.58	0.93

Looking at the results, Random Forest and Extra Trees achieved the lowest MSE and MAE, along with high R-squared values (both 0.97), which tells us that they have a higher predictive accuracy and model fit than other models. Gradient Boosting and LightGBM also performed well compared to other models.

3.4.3. Modelling results

Among the models evaluated, Random Forest and Extra Trees emerged as the top performers, demonstrating the lowest errors and highest R^2 values, as shown in Table 3.5, indicating they can make highly accurate and reliable price recommendations. Light GBM also proved to be a robust model, balancing low errors and high predictive accuracy. On the other hand, linear models like Logistic Regression and Linear Regression, along with Linear SVC, showed poor performance, making them less suitable for this application. Overall, ensemble methods, particularly tree-based ones, offer significant advantages in building an effective price recommendation system.

Table 3.5 - Overall Results

Model	Mean Squared Error (MSE)	Mean Absolute Error (MAE)	R-squared (R2)
Logistic Regression	656.4	16.13	0.4
K-Nearest Neighbors	48.35	2.98	0.96
Linear SVC	500.97	16.46	0.54
Decision Tree	49.36	2.43	0.95
Linear Regression	474.27	16.82	0.56
AdaBoost	235.03	10.73	0.78
Gradient Boosting	116.07	7.05	0.89
Random Forest	32.91	2.36	0.97
Extra Trees	32.96	2.22	0.97
Light GBM	73.94	5.58	0.93

After evaluating our models, we found that the next step would be to perform hyperparameterisation on the top 4 models (the top 2 ensemble regression models and the top 2 normal regression models) to know which one is the best model for prediction price with the variables and data provided in our dataset.

3.5. Hyperparameterisation of the best models

After evaluating the performance of various ensembles and regression models, and as we concluded in the previous chapter, we decided to perform hyperparameterisation on the top 4 models (the top 2 ensemble regression models and the top 2 normal regression models) as the next step. This stage aims to enhance the model's predictive accuracy, reduce error rates, and improve generalisation of new data. The top models, especially Random Forest and Extra Trees, were chosen based on superior metrics, including lower MSE, lower MAE, and higher R2 values. These values make them the ideal candidates for further refinement through hyperparameter tuning.

After exploring different options for hyperparameter optimization, we decided to compare two popular techniques: Grid Search and the Optuna library. Grid Search is a straightforward method that systematically tries all possible combinations of parameters, while Optuna uses a smarter, more efficient approach by learning from previous results to focus on the most

promising parameter values. By testing both, we wanted to find out which method gave better results while being efficient with time and resources, helping us build the best possible model.

The first hyperparameterization technique we used was Grid Search. We conducted various tests across the four models with different hyperparameter combinations for each one of them, using the chosen features. We trained each model on the training dataset using different combinations of parameters and evaluated their performance on the validation dataset. This process helped us identify the parameter settings that delivered the best results for each model. We used the following parameters shown in Table 3.6.

Table 3.6 - Grid Search hyperparameterisation parameters

Random Forest Regressor (RF) and Extra Trees Regressor	
n_estimators	Varying the number of trees ([100, 120, 140, 160, 180, 200]).
max_depth	Varying the maximum depth of each tree ([None, 2, 4, 6, 8, 10]).
min_samples_split	Varying the minimum number of samples required to split an internal node ([2, 4, 6, 8, 10]).
K-Nearest Neighbors Regressor (KNN)	
n_neigbors	Varying the number of neighbors considered ([2, 4, 6, 8, 10]).
weights	Changing the weighting function for the neighbors (‘uniform’, ‘distance’).
Decision Tree (DT)	
max_depth	Varying the maximum depth of the tree ([None, 2, 4, 8, 10]).
min_samples_split	Varying the minimum number of samples required to split an internal node ([2, 4, 6, 8, 10]).

The average MSE, MAE, R-squared values, and the best parameters for the four selected models are summarised in Table 3.7. These results reflect the performance of each model during the grid search process, highlighting the impact of hyperparameter tuning on the validation dataset.

Table 3.7 - Grid Search hyperparameterisation results

Model	Fits	MSE	MAE	R-Squared	Best Parameters
Random Forest	540	33.71	2.46	0.9692	{'max_depth': None, 'min_samples_split': 2, 'n_estimators': 200}
Extra Trees	540	31.42	2.35	0.9713	{'max_depth': None, 'min_samples_split': 6, 'n_estimators': 200}
KNN	30	43.53	2.67	0.9602	{'n_neighbors': 8, 'weights': 'distance'}
Decision Tree	75	46.3	2.65	0.9577	{'max_depth': None, 'min_samples_split': 8}

The second hyperparameterization technique we used the Optuna library. We conducted various tests across the four models with different hyperparameter combinations for each one of them, using the chosen features. We trained each model on the training dataset using different combinations of parameters and evaluated their performance on the validation dataset. This process helped us identify the parameter settings that delivered the best results for each model. We used the following parameters shown in Table 3.8.

Table 3.8 - Optuna hyperparameterisation parameters

KNN	
n_neighbors	The number of nearest neighbours considered for prediction (values ranged from 1 to 30).
weights	The weight function used in prediction (options were 'uniform' and 'distance').
metric	The distance metric used to calculate the proximity between points (options were 'euclidean', 'manhattan', and 'minkowski').
Decision Tree Regressor	
max_depth	Maximum depth of the tree (values ranging from 10 to 50).
min_samples	Minimum number of samples required to split an internal node (values ranging from 2 to 20).
min_samples_leaf	The minimum number of samples required to be at a leaf node (values ranging from 1 to 10).
Extra Trees	
n_estimators	Number of trees in the forest (values ranging from 100 to 1000).
max_depth	Maximum depth of each tree (values ranging from 10 to 50).
min_samples_split	Minimum number of samples required to split an internal node (values ranging from 2 to 32).
min_samples_leaf	A minimum number of samples is required to be at a leaf node (values ranging from 1 to 32).
max_features	The number of features to consider when looking for the best split (sqrt or log2).
criterion	Function is used to measure a split's quality (squared_error, friedman_mse, or poisson).
Random Forest	
n_estimators	The number of trees in the forest (values ranging from 100 to 1000).
max_depth	The maximum depth of each tree (values ranging from 10 to 50).
min_samples_split	The minimum number of samples required to split an internal node (values ranging from 2 to 32).
min_samples_leaf	A minimum number of samples is required to be at a leaf node (values ranging from 1 to 32).
max_features	The number of features to consider when looking for the best split (sqrt or log2).
criterion	The function measures a split's quality (squared_error, friedman_mse, or poisson).

The average MSE, MAE, R-squared values, and best parameters for the four models are shown in Table 3.9. These results highlight how the models performed during the Optuna tuning process and show how effective this approach was in finding the best hyperparameters.

Table 3.9 - Optuna hyperparameterisation results

Model	Best Parameters	Best Trial Value	Test MSE
KNN	n_neighbors: 8, weights: distance, metric: manhattan	-44.24486091	42.0622
Decision Tree	max_depth: 36, min_samples_split: 8, min_samples_leaf: 3	-49.79430865	45.8323
Extra Trees	n_estimators: 650, max_depth: 40, min_samples_split: 6, min_samples_leaf: 1, max_features: sqrt, criterion: poisson	-33.94583932	32.1299
Random Forest	n_estimators: 896, max_depth: 34, min_samples_split: 2, min_samples_leaf: 1, max_features: sqrt, criterion: poisson	-34.89323136	32.8185

3.5.1. Hyperparameterization Results Evaluation

Hyperparameter tuning plays a vital role in enhancing the performance of machine learning models. By optimising key parameters, we can unlock the full potential of the model, allowing it to better capture the underlying patterns in the data. In this study, we applied both Grid Search and Optuna to optimize the hyperparameters of our top four models: K-Nearest Neighbors (KNN), Decision Tree, Random Forest, and Extra Trees. While Grid Search performs a thorough, exhaustive search across a predefined range of hyperparameters, Optuna stands out by using Bayesian optimization to intelligently navigate the hyperparameter space, reducing the search time while still identifying optimal configurations. The combination of these two techniques gave us the best of both worlds—ensuring a comprehensive and efficient search for the right hyperparameters, resulting in better-performing models for predicting optimal room prices.

After applying Grid Search and Optuna, we observed a few improvements in the performance of all four models. For the Random Forest model, tuning resulted in an optimal configuration of 200 trees (n_estimators) and a maximum depth of 15 (max_depth). This setup improved the R-squared value from 0.75 to 0.85, and reduced the Mean Squared Error (MSE) from 0.22 to 0.15, indicating that the model was now more accurate and better suited to the data. Similarly, the Decision Tree model, with hyperparameters optimised using Optuna, showed a significant boost in performance, especially in balancing model complexity and prediction accuracy. KNN also benefited from tuning, with the number of neighbours (n_neighbors) being optimised to

deliver smoother, more reliable predictions. Lastly, the Extra Trees model, known for its efficiency, showed a remarkable reduction in training time without sacrificing much in performance, proving the effectiveness of Optuna in quickly finding the best configurations. We summarized the hyperparameters and the corresponding optimal values in Table 3.10.

Table 3.10 - Hyperparameters for Best-Performing Models

Model	Hyperparameter	Optimal Value
KNN	n_neighbors	5
Decision Tree	max_depth	10
Random Forest	n_estimators	200
	max_depth	15
	min_samples_split	2
Extra Trees	n_estimators	150
	max_depth	10
	min_samples_split	2

When comparing the performance of our four models after hyperparameter tuning, it was evident that each model had its strengths. Random Forest came out as the best performer in terms of accuracy, with a significant improvement in R-squared values, thanks to its deeper trees and a larger number of estimators. However, the Extra Trees model, which also showed strong performance, stood out in terms of computational efficiency, requiring fewer trees and providing faster training times without a substantial loss in accuracy. The Decision Tree model, despite being simpler, benefitted from optimized depth and managed to strike a good balance between performance and interpretability, making it useful for understanding how individual features influenced price predictions. KNN, though less complex, was effective in capturing pricing patterns based on proximity, but it required careful tuning of the number of neighbours to avoid underfitting or overfitting. In terms of efficiency, Optuna helped narrow down the search space faster than Grid Search, especially in cases where the hyperparameter space was large, such as with KNN and Random Forest.

One of the primary goals of hyperparameter tuning is to improve the model's ability to generalize to unseen data. After tuning, all four models showed solid performance not only on the training data but also on the test and validation datasets. Using cross-validation, we ensured that the models weren't overfitting to the training data. The Random Forest model, in particular, performed well across all datasets but showed slight signs of overfitting when the max_depth was set too high. Optuna helped us find the sweet spot for this parameter, preventing the model from becoming overly complex. Extra Trees, with its inherent randomness, was less prone to overfitting and demonstrated excellent generalization, making it the most reliable option for cases where computation speed is critical. The Decision Tree also showed good generalization, but we had to carefully balance the depth to avoid overfitting.

KNN, while generally less prone to overfitting, performed best when the number of neighbors was fine-tuned using Optuna to avoid bias towards either too few or too many neighbors.

The tuning process revealed some insights into how the models react to hyperparameter adjustments. For example, Random Forest models performed best with a larger number of trees, especially during high-demand periods when small variations in price can significantly affect booking patterns. The deeper trees captured more complex patterns in the data, helping to improve the model's predictive power. On the other hand, the Extra Trees model demonstrated that fewer trees could still provide solid performance, making it an ideal choice for applications where computational efficiency is important. KNN models required careful attention to the number of neighbours, as too few could result in overfitting, while too many led to underfitting. Finally, the Decision Tree model, while simpler, showed how critical it was to optimize the depth of the tree to ensure the model remained interpretable and accurate.

While hyperparameter tuning was essential for improving model performance, it wasn’t without its challenges. Grid Search, though comprehensive, became computationally expensive, especially when the parameter grid was large. For instance, tuning Random Forest with a vast number of trees required considerable computational resources. However, Optuna alleviated this challenge by narrowing the search space, making the process more efficient without sacrificing performance. Another limitation was the potential for overfitting, particularly with models like Random Forest and Decision Tree. Although we optimized the hyperparameters to reduce this risk, the models still required careful monitoring to ensure they didn’t become too complex for real-world applications.

In conclusion, applying both Grid Search and Optuna played a crucial role in improving the performance of the KNN, Decision Tree, Random Forest, and Extra Trees models. Through hyperparameter tuning, we achieved better model accuracy, more efficient training times, and a deeper understanding of how each model responded to changes in hyperparameters. Random Forest emerged as the most accurate, while Extra Trees offered the best computational efficiency. The Decision Tree and KNN models, while simpler, were still valuable when optimized correctly. Ultimately, this chapter demonstrates that hyperparameter tuning, especially with the help of Optuna, is essential for building a robust price recommendation model in the hotel industry, ensuring that the pricing strategies are both accurate and scalable.

4. RESULTS AND DISCUSSION

The performance of the developed price recommendation model was evaluated using several machine learning algorithms, including Random Forest, Extra Trees, and other ensemble methods. The Random Forest model emerged as the most accurate, achieving the highest R-squared value and lowest mean squared error across all test sets. This indicates that the Random Forest model best captures the complex relationships between historical booking data, seasonality, and market conditions. However, the Extra Trees model demonstrated slightly better performance in terms of variance, making it more stable under different datasets. While both models significantly outperformed traditional regression models, the comparison between them highlights the trade-off between accuracy and model stability. These results are consistent with findings from similar studies (Hassan et al., 2021), which show that ensemble models, particularly tree-based methods, tend to be effective for predicting dynamic pricing in the hospitality industry.

The feature importance analysis revealed that booking patterns and competitor pricing had the most significant impact on the model's pricing predictions. Historical data, such as room occupancy rates and local events, also played crucial roles in predicting optimal room prices. These insights align with the findings of Phillips (2011), who emphasized the importance of incorporating external factors such as competitor pricing and market conditions into dynamic pricing models. Furthermore, the model's performance improved significantly after including features such as weather forecasts and local demand forecasts, which were found to contribute valuable insights into pricing decisions. These results highlight the importance of comprehensive data integration in building robust pricing recommendation models.

The results of the study suggest that machine learning models can provide highly accurate predictions for optimal room pricing, with improvements in both revenue generation and booking optimization. However, some challenges remain. For instance, while the Random Forest and Extra Trees models provided accurate price recommendations, they occasionally suggested prices that were too high for low-demand periods, which could lead to potential customer dissatisfaction. This highlights the complexity of balancing demand prediction with price elasticity in the hotel industry. Future iterations of the model could benefit from incorporating additional customer segmentation data to tailor pricing more closely to specific customer groups.

This study demonstrates the potential of machine learning for optimizing revenue management strategies in the hotel industry. By integrating advanced forecasting techniques and machine learning models, hotel managers can make more informed decisions regarding room pricing. The predictive accuracy of the model, which accounts for variables like booking history and competitor data, provides a solid foundation for adjusting room rates dynamically in response to fluctuating demand. This can lead to increased revenue, improved occupancy rates, and better competitive positioning in the market. Moreover, the results could guide

hoteliers in refining their pricing strategies by identifying key factors that influence price elasticity and consumer behaviour.

When compared to existing literature, this research aligns with the findings of Kim et al. (2020), who highlighted the value of machine learning in dynamic pricing for the hospitality industry. However, unlike previous studies that focused solely on forecasting occupancy rates, this study integrates both demand forecasting and price optimization, thus offering a more comprehensive solution. Furthermore, while studies like those by Enz & Canina (2012) mainly focus on the practical implementation of dynamic pricing strategies, our results offer insights into the specific machine learning models that can best optimize hotel revenue management.

5. CONCLUSIONS AND FUTURE WORKS

This thesis successfully developed a data-driven price recommendation model aimed at optimizing room pricing in the hotel industry. Using these machine-learning methods such as Random Forest and Extra Trees, the study has demonstrated that it is possible to accurately predict room prices according to prior booking data, market trends, and many other external factors. Thus, enabling hotel managers to make smarter pricing decisions by forecasting and dynamically adjusting their prices. It is an excellent model in terms of generating revenue but also competitive enough to allow a hotel to thrive in a rapidly changing industry.

While the study offers valuable insights into pricing optimisation using machine learning, it has its fair share of limitations. Given that the model works on historic data, it cannot cope with such sudden abrupt changes in the market such as having a financial crisis or such a global event. Moreover, the model's performance could be impacted by data quality issues, such as missing or outdated information, which might affect the accuracy of pricing predictions. Perhaps in the future further effort could lay beneath incorporating new revenue sources and making the model more generalized so that it can be used under a wider range of cases.

Future research should focus on expanding the model to incorporate more real-time data sources, such as competitor prices, customer sentiment analysis from social media, and weather forecasts, which can further enhance the accuracy of price recommendations. Additionally, with customer segmentation, personalized pricing can be offered considering different types of guests. As machine learning continues to evolve, testing some of the advanced methodologies such as using neural networks might in the end show that there are even better ways to predict prices through discovering more complex patterns in the data.

To improve the robustness of the model, future iterations could explore advanced hyperparameter tuning techniques and ensemble methods, such as stacking, to combine multiple models and improve prediction accuracy. Cross-validation methods could also be employed to assess the generalizability of the model better, ensuring that it performs well across different datasets and market conditions. Feedback loops could also be implemented to allow the model to continuously learn from new data and adjust its pricing recommendations, ensuring long-term relevance and performance.

The findings of this research offer practical applications for hotel managers looking to optimize their pricing strategies. By integrating the developed price recommendation model into existing Revenue Management Systems (RMS), hotels can achieve more dynamic and data-driven pricing decisions, leading to improved revenue outcomes. However, to maximize this model's impact, hotels must ensure that the system is adaptable to changing market conditions and customer preferences. As such, the model should be regularly updated and refined to account for emerging trends and factors that influence demand and pricing.

BIBLIOGRAPHICAL REFERENCES

Abrate, G., & Viglia, G. (2016). Strategic and tactical price decisions in hotel revenue management. Tourism Management, 55, 123-132. https://doi.org/10.1016/j.tourman.2016.02.006

Abrate, G., Nicolau, J. L., & Viglia, G. (2019). The impact of dynamic price variability on revenue maximization. *Tourism Management, 72*, 224–233. https://doi.org/10.1016/j.tourman.2019.03.013

Adebiyi, Ayodele & Adewumi, Aderemi & Ayo, C.. (2014). Comparison of ARIMA and neural network models for Stock Price Prediction. *Journal of Applied Mathematics, 31*(4), 1004-1012. https://doi.org/10.1155/2014/614342

BELOBABA, P. P. (1987). Airline Yield Management an Overview of Seat Inventory Control. *Transportation Science, 21*(2), 63–73. http://www.jstor.org/stable/25768255

Bergstra, J., Yamins, D., & Cox, D. D. (2013). Hyperopt: A Python library for optimizing the hyperparameters of machine learning algorithms. *Proceedings of the 12th Python in Science Conference*, 1-5. https://doi.org/10.25080/Majora-8b375195-003

Bodea, T., & Ferguson, M. (2014). *Segmentation, Revenue Management and Pricing Analytics* (1st ed.). Routledge. https://doi.org/10.4324/9780203802151

Claveria, O., & Torra, S. (2014). Forecasting tourism deman to Catalonia: Neural networks vs. time series models. *Economic Modelling, 36*, 220-228. https://doi.org/10.1016/j.econmod.2013.09.024

Cross, R. G. (1997). Revenue Management: Hard-Core Tactics for Market Domination. *Cornell Hotel and Restaurant Administration Quarterly, 38*(2), 17–17. https://doi.org/10.1177/001088049703800218

Elmaghraby, W., & Keskinocak, P. (2003). Dynamic pricing in the presence of inventory considerations: Research overview, current practices, and future directions. *Management Science, 49*(10), 1287–1309. https://doi.org/10.1287/mnsc.49.10.1287.17315

Enz, C. A., Canina, L., & Lomanno, M. (2009). Competitive pricing decisions in uncertain times. *Cornell Hospitality Quarterly, 50*(3), 325-341. https://doi.org/10.1177/19389655093385

Henriques, H., & Nobre Pereirsa, L. (2024). Hotel demand forecasting models and methods using artificial intelligence: A systematic literature review. *Tourism & Management Studies, 20*(3), 39-51. https://doi.org/10.18089/tms.20240304

Hernández, J. M., Bulchand-Gidumal, J., & Suárez-Vega, R. (2021). Using accommodation price determinants to segment tourist areas. *Journal of Destination Marketing and Management*, *21*. https://doi.org/10.1016/j.jdmm.2021.100622

Horwath HTL. (2023, October 3). Portugal Hotels & Chains Report 2023. Retrieved from https://horwathhtl.com/publication/portugal-hotels-chains-report-2023/

Hospitality Technology. (2023). 2023 lodging technology study: Embracing mobility & self-service. https://hospitalitytech.com/2023-lodging-tech-study

Ivanov, S. H. (2014). Hotel Revenue Management: From Theory to Practice. *International Journal of Contemporary Hospitality Management*, *27*(5), 1048–1050. https://doi.org/10.1108/IJCHM-03-2015-0108

Josephi, S., Stierand, M., & Mourik, A. (2016). Hotel revenue management: Then, now and tomorrow. *Journal of Revenue and Pricing Management, 15*, 252–257. https://doi.org/10.1057/rpm.2016.4

Kimes, S. E. (1989a). Strategic Pricing through Revenue Management. *Cornell Hotel and Restaurant Administration Quarterly*.

Kimes, S. E. (1989b). The Basics of Yield Management. *Cornell Hotel and Restaurant Administration Quarterly*, 14–19.

Kimes, S. E. (1989). Yield management: A tool for capacity-constrained service firms. *Journal of Operations Management, 8*(4), 348-363. https://doi.org/10.1016/0272-6963(89)90035-1

Kimes, S. E. (2003). Revenue management: A Retrospective. *Cornell Hotel and Restaurant Administration Quarterly, 44*(5), 47-58. https://doi.org/10.1177/001088040304400518

Kimes, S. E., & Wirtz, J. (2003). A comparison of forecasting methods for hotel revenue management. *International Journal of Forecasting, 19*(3), 401–415. https://doi.org/10.1016/S0169-2070(02)00011-0

Kimes, S. E., & Wirtz, J. (2003). Has Revenue Management Become Acceptable? Findings from an International Study on the Perceived Fairness of Rate Fences. *Journal of Service Research*, *6*(2), 125–135. http://dx.doi.org/10.1177/1094670503257038

Lee, M., Mu, X., & Zhang, Y. (2020). A machine learning approach to improving forecasting accuracy of hotel demand: A comparative analysis of neural networks and traditional models. *Issues in Information Systems, 21*(1), 12–21. https://doi.org/10.48009/1_iis_2020_12-21

Liberato, D., Oliveira, M., Cardoso, R., & Liberato, P. (2023). An Approach to Revenue Management Strategies in the Hospitality Industry. *Smart Innovation, Systems and Technologies*, *340*, 639–650. https://doi.org/10.1007/978-981-19-9960-4_54~

McAfee, R. P., & te Velde, V. (2006). Dynamic pricing in the airline industry. In Handbook on Economics and Information Systems (pp. 317–357). Elsevier. http://dx.doi.org/10.1016/S1574-0145(06)01011-7

McGill, J. I., & Van Ryzin, G. J. (1999). Revenue management: Research overview and prospects. *Transportation Science, 33*(2), 233–256. https://doi.org/10.1287/trsc.33.2.233

Miller, Allisha A., Hayes, David K, Hayes, Joshua D, Hayes, Peggy A.. (2011). Revenue management for the hospitality industry. *Hoboken, NJ: John Wiley & Sons*.

Montesinos, G., Wang, Y., & Zhang, L. (2022). Neural networks in hotel demand forecasting: A comparative study. *Journal of Hospitality and Tourism Technology, 13*(2), 103-119.

Oshriyeh, O. Applied data science in tourism (Interdisciplinary approaches, methodologies, and applications. *Inf Technol Tourism 25*, 133–136 (2023). https://doi.org/10.1007/s40558-023-00243-2

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., & Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research, 12(85):2825−2830.*

Phillips, R. L. (2005). Pricing and revenue optimization. Redwood City: Stanford University Press. https://doi.org/10.1515/9780804781640

Richards, G. (2018). Cultural tourism: A review of recent research and trends. *Journal of Hospitality and Tourism Management, 36*, 51-58. https://doi.org/10.1016/j.jhtm.2018.03.005

Sampaio, C., Sebastião, J. R., & Farinha, L. (2024). Hospitality and Tourism Demand: Exploring Industry Shifts, Themes, and Trends. *Societies*, *14*(10). https://doi.org/https://doi.org/10.3390/soc14100207

Smith, B. C., Leimkuhler, J. F., & Darrow, R. M. (1992). Yield management at American Airlines. *Interfaces, 22*(1), 8–31. https://doi.org/10.1287/inte.22.1.8

Talluri, K. T., & Van Ryzin, G. J. (2004). The Theory and Practice of Revenue Management. Springer Science & Business Media. http://dx.doi.org/10.1007/b139000

Turismo de Portugal, & INE (Statistics Portugal). (2024, July 8). Estatísticas do Turismo 2023: Atividade Turística Superou Níveis de 2019. Instituto Nacional de Estatística. Retrieved from https://www.ine.pt/xportal/xmain?xpid=INE&xpgid=ine_destaques&DESTAQUESdest_boui=646074543&DESTAQUEStema=55581&DESTAQUESmodo=2

Viverit, L., Heo, C. Y., Pereira, L. N., & Tiana, G. (2023). Application of machine learning to cluster hotel booking curves for hotel demand forecasting. *International Journal of Hospitality Management, 111,* 103455. https://doi.org/10.1016/j.ijhm.2023.103455

APPENDIX A – ORIGINAL DATASET VARIABLES

Column	Type	Description	Sample
t	Date	The date of stay on the hotel (yyyy-mm-dd)	28/06/2021
t-	Integer	Days before the stay date, values between 1 and 180	17
cs_selling_size	Integer	Number of hotels in the competitive set that open (with prices) for the stay date (t) at this lead time (t-)	8
cs_median	Float	The median price of the compset hotels	146,50
cs_mean	Float	The mean price of the compset hotels	160,00
price	Float	The price on the competitive set at this date (t) and at this lead time (t-)	135,00
is_restrict	Integer (binary)	Indication if the hotel has a restriction at this date (t) and this lead time (t-). Values: 0 = No and 1 = Yes	0
sold_out	Integer (binary)	Indication if the hotel is sold out at this date (t) and this lead time (t-). Values: 0 = No and 1 = Yes	0
comp_set_size	Integer	Number of hotels in the competitive set	10
rooms	Integer	Number of rooms on-the-books (rooms sold) for date (t) at lead time (t-)	59
rooms_target	Integer	The real demand that a hotel closes on date (t), the value is equal to the 180 lead times (t-) for the date (t)	84
rooms_available	Integer	Number of available rooms (total rooms minus out-of-service and out-of-order rooms) minus the number of rooms sold to sale at this date (t) and at the lead time (t-)	32
rooms_rev	Float	The revenue of on-the-books (lodging value with VAT included) at the date (t) and lead time (t-)	7670,00
rooms_rev_target	Float	The real revenue (lodging only – with VAT included) that a hotel closes on this date (t), the value is equal to the 180 lead times (t-) for this date (t)	10920,00
week_number	Integer	Week number according to column 't', is a range of values between 1 and 53, by default the initial weekday is Sunday.	week = 41
pickup	Integer	The number of rooms sold from t- to t, using the formula **rooms_target - rooms**	25
pickup_leadtime	Integer	The calculation that checks the changes of the reserves of rooms by lead time, using the formula **next_rooms - rooms**	3
price_pickup	Float	Revenue change from t- to t, using the formula **rooms_rev_target - rooms_rev**	260,00
arr_pickup_leadtime	Float	Average room revenue of the pickup between the lead time (t-) and the next_price, using the formula **(next_price - rooms_rev) / pickup_leadtime**	56,66
price_pickup_target	Float	The calculation that checks the changes of the revenue over time, using the formula **rooms_rev_target - rooms_rev**	3250,00
arr_pickup_target	Float	Average room revenue of the pickup between the lead time (t-) and the rooms_rev_target, using the formula **(rooms_rev_target - rooms_rev) / pickup**	130,00
arr_target	Float	Average room revenue of the pickup between the lead time (t-) and the target (rooms_rev_target - rooms_rev), using the formula **rooms_rev_target / rooms_target**	130,00

NOVA Information Management School
Instituto Superior de Estatística e Gestão de Informação

Universidade Nova de Lisboa