UMinho | 2023

Adriana Pinho

A Data Analysis Approach to Evaluate the Performance of Predictive Models

University of Minho
School of Engineering

Adriana Costa Pinho

A Data Analysis Approach to Evaluate the Performance of Predictive Models

october 2023

University of Minho
School of Engineering

Adriana Costa Pinho

A Data Analysis Approach to Evaluate the Performance of Predictive Models

Master Thesis
Master in Systems Engineering

Dissertation supervised by
Professora Doutora Ana Maria Alves Coutinho da Rocha
Professor Doutor Manuel Carlos Barbosa Figueiredo

october 2023

Copyright and Terms of Use for Third Party Work

License granted to users of this work:

Acknowledgements

To Professors Ana Maria Alves Coutinho Rocha and Manuel Carlos Barbosa Figueiredo, for accepting the challenge of being the supervisors of this dissertation. To my parents and my boyfriend, for always trusting me.

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 Code of Ethical Conduct of the University of Minho.

Abstract

A data analysis approach to evaluate the performance of predictive models

Bikes have become an increasingly popular mode of transportation, mainly due to their agility in covering short distances and as a sustainable mode of transportation. More and more cities worldwide are adopting bike-sharing systems, where users can rent a bike for a small fee.

However, this shift also brings its own challenges, that ranges from ensuring safety and robust infrastructure to managing costs. One prominent issue is ensuring bike availability. After all, the main value proposition of such systems is their convenience. If a user approaches a rental station and do not finds any bike available, or if there are no slots to return the rented bike, the system loses its purpose. Predicting the number of rentals each station will have each day is a challenge, as this value is highly unpredictable and can be influenced by various factors, from the weather to local events.

This dissertation addresses the bike availability issue in a bike-sharing system. Having bikes available at the right place and the right time is the key to success. To effectively forecast demand, two predictive algorithms were implemented: SARIMAX and *Gradient Boosting*. SARIMAX is a variant of the well-known ARIMA, recognized for its accuracy in time series forecasting. On the other hand, *Gradient Boosting*, an algorithm based on decision-trees, is widely used because of its ability to handle vast amounts of data with minimal computational resources.

The core question of this dissertation is to determine which of these algorithms will best predict the daily demand of each station, ensuring that users always have a bike available when and where they need it. This guarantees users satisfaction and, in return, promotes the growth of companies managing such systems. Based on the daily rental volumes of each station, the *Gradient Boosting* algorithm was the one that presented the best performance. This performance was further improved when the stations were divided into clusters, depending on their rental volume.

Keywords: Bike-Sharing System, Demand Forecasting, *Gradient Boosting*, ARIMA, SARIMA, SARIMAX

Resumo

Uma abordagem de análise de dados para avaliar o desempenho de modelos preditivos

As bicicletas têm-se tornado um meio de transporte cada vez mais popular, principalmente devido à sua flexibilidade para percorrer curtas distâncias e ao facto de serem um meio de transporte sustentável. Cada vez mais as cidades estão a adotar sistemas de partilha de bicicletas, onde os utilizadores podem alugar uma bicicleta mediante o pagamento de uma pequena taxa.

No entanto, essa mudança traz os seus próprios desafios, que variam desde a garantia de segurança e infraestruturas robustas até à gestão de custos. Um problema proeminente é garantir a disponibilidade de bicicletas. Afinal, o principal benefício deste tipo de sistemas é a sua conveniência. Se um utilizador se desloca a um estação de aluguer e não encontra nenhuma bicicleta disponível, ou se não há vagas disponíveis para devolver a bicicleta alugada, o sistema perde o seu propósito. Prever o número de alugueres que cada estação terá em cada dia é um desafio, uma vez que este valor é altamente imprevisível, podendo ser influenciado por vários fatores, desde o clima até eventos locais.

Esta dissertação aborda a questão da disponibilidade de bicicletas num sistema de partilha de bicicletas. Ter bicicletas disponíveis no lugar e momento certo é a chave para o sucesso. Para prever a procura de forma eficaz, foram implementados dois algoritmos de previsão: SARIMAX e *Gradient Boosting*. SARIMAX é uma variante do conhecido ARIMA, reconhecido pela sua precisão na previsão de séries temporais. Por outro lado,o *Gradient Boosting*, um algoritmo baseado em árvores de decisão, é amplamente utilizado pela sua capacidade de lidar com grandes volumes de dados utilizando poucos recursos computacionais.

A questão central desta dissertação é determinar qual destes algoritmos prevê melhor a procura diária de cada estação, garantindo que os utilizadores têm sempre uma bicicleta disponível quando e onde necessitam. Com base nos volumes diários de aluguer de cada estação, o algoritmo *Gradient Boosting* foi o que apresentou melhor desempenho, que foi ainda melhorado quando as estações foram divididas em clusters com base no seu volume de alugueres.

Palavras-chave: Sistema de Partilha de Bicicletas, Previsão da Procura, *Gradient Boosting*, ARIMA, SARIMA, SARIMAX

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Almost all urban areas are growing in population. Therefore, the need for efficient, sustainable, and user-friendly public transportation systems becomes of extreme importance. Transportation not only facilitates the movement of individuals to access the opportunities offered by cities but also mirrors the socioeconomic dynamics and health of a city (Hernández 2017). Some of the most populated urban areas in the world have impressive public transportation systems, like Hong Kong, Tokyo or Berlin (Bills 2023).

There are multiple modes of public transportation, such as trains, subways, bus, trams, boats, taxis, public bikes or scooters, and so on. This dissertation will focus on bike-sharing systems.

1.1 Context

According to the United Nations, more than 55% of the world's population lives in urban areas and it is expected that these percentage increases to almost 70% in the next 30 years (UnitedNations 2023). This migratory flow leads to challenges for the cities. One of the main concerns about that is the lack of infrastructures, especially regarding to urban mobility. The increase of population density intensifies the traffic, which brings problems such as congestion, accidents, pollution and long time travel. This problems not only affects life quality, but also creates barriers to the economic growth of the area, not to mention the increasingly evident environmental crisis.

This way, it emerges the need of alternative and sustainable ways of transportation. On a daily basis, a significant number of people use public transports as the only or main kind of transportation, like train, subway, bus, etc. Although it reduces the amount of vehicles on the streets, which inherently brings substantial advantages, it does not give the flexibility of private modes of transportation, like private cars. That flexibility can be achieved through public bikes, for example, and there are an increasingly growing number of cities that are adopting bike-sharing systems as a way of reducing pressure on public and private transportation systems, reducing the ecological footprint and promoting a healthy and sustainable life style. This public bike-sharing systems allows people to rent a bike in one specific location and deliver it in the same or another location, by paying a small fee.

The first bike-sharing systems emerged in the 1960s in the Netherlands, but it was only in the 2000s that they began to grow (DeMaio 2003), being a very popular mode of transportation nowadays.

1.2 Problem

Managing a bike-sharing system is a challenge. That is why there where numerous generations of these systems that evolved over time, with each iteration becoming progressively more refined than its predecessor. But for these enhancements, it was necessary to learn from mistakes.

Some of the major concerns about managing a bike-sharing system are:

- Distribution and reallocation of bikes: It is essential that bikes are available at the right time at the right station. For that, it is fundamental to predict the amount of bikes each station is going to need every day. Furthermore, having an excess of bikes at some stations leads to costs, while a deficit of bikes at other stations results in lost sales. Therefore, maintaining the right number of bikes at each station is a crucial activity. This right number should allow stations to always have bikes available to people, but without excessive bikes for lack of demand;

- Infrastructure: Riding a bike can be dangerous. In 2019, 9% of all roads fatalities were cyclists (European Commission 2021). In order to prevent accidents, appropriate roads and bike lanes are essential. The challenge is to create this infrastructure without creating inconveniences for other existing types of transportation. Additionally, these systems require parking and specific road signs. Therefore, when it is intended to implement such systems, it is essential to realize whether the city already has or allows it to have dedicated infrastructure to avoid accidents and embarrassment with other transports;

- Integration with other types of transports: Bikes are often used as first/last mile transport, in connection with other types of transportation. In order to facilitate this connectivity, the stations must be placed at the right place;

- Security: The first bike-sharing system failed because, among others, the bikes were stolen (DeMaio 2003). When implementing this systems, it is necessary that the whole system needs to be theft resistant. Furthermore, the bikes need to be safe for the people who will be using them, being equipped with features and characteristics that increase cyclist safety;

- Software and Technology: One of the main successes of almost all digital businesses is having a user friendly software. For this systems, the platform must clearly inform the location of the bikes and should have an easy to use payment methods. In addition, the method of renting a bike should be simple, otherwise they may lose customers;

- Culture and environmental challenges: When introducing this systems for the first time in new cities, it is important to create marketing campaigns suited to the population in order to attract customers. In addition, meteorological conditions can be a challenge in some places. Solutions must be designed to ensure that the system remains attractive and operational, during adverse weather conditions such as periods of snow or cities immersed in pollution-induced fog.

Despite the various challenges listed, this dissertation will only focus on the first topic listed above.

1.3 Objective

In this industry, there are moments of high demand contrasted with times of lower demand. An excess or deficit of bikes at specific stations during particular times results in costs for enterprises, along with customer dissatisfaction.

To ensure the correct distribution and reallocation of bikes, it is imperative to have an in-depth knowledge about the rentals of all stations. In order to achieve this, descriptive analyses should be carried out on how external variables influence the demand for bike rental, or the identification of the most frequented stations or even peak usage times during the day or week. Knowing which stations are most or less used, for example, is a crucial factor to determine the amount of bikes each station needs. Other than that, and knowing the surroundings, determine whether it is necessary to create new infrastructures. This information is also important for guiding the appropriate marketing campaigns for each station.

Broadly speaking, the objective of this research is to predict the daily demand for bike rental for each station. Various factors, including weather conditions, fuel prices and holidays will be taken into account to discern their influence on the forecasts.

These forecasts are important because, among others, they help bike-sharing companies to optimize the allocation of bikes/software/infrastructure/human resources, contributing to a better operational efficiency. It also helps to prepare for upcoming events, such as local events or natural disasters. Taking into account the consumer's perspective, what they want is to know that, no matter the time of the day, they will have a bike available at the right station, at the time they need it and at an affordable price. Satisfied customers are the main key to the long-term success of companies.

Holistically, forecasting these rentals optimizes the management of these companies, allowing funds to be channeled towards value-driven initiatives while data-driven decision-making leads to better results.

1.4 Work Methodology

This project follows the Cross-Industry Standard Process for Data Mining (CRISP-DM) methodology. The CRISP-DM is a robust and well-established methodology that provides a structured approach to planning and executing data mining projects (Schröer, Kruse, and Gómez 2021).

Among other aspects, it stands out from other methods by recognizing the iterative nature of data mining. The process is rarely linear, meaning that moving back to earlier phases is often necessary as more is understood about the data and the problem. Moreover, teams which were trained and explicitly told to implement CRISP-DM had better performance than teams using other approaches (Saltz, Shamshurin, and Crowston 2017).

CRISP-DM breaks down the data mining process into six phases:

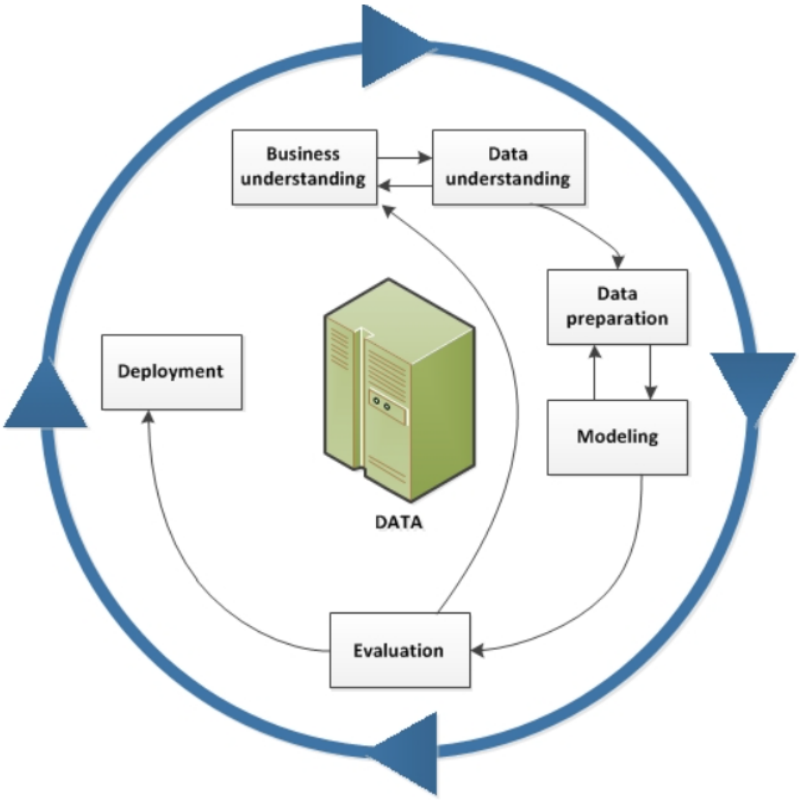


Figure 1. CRISP-DM: The data mining life cycle (IBM 2023)

Business Understanding: This phase aims to understand the requirements of the business. Some tasks include: discovering the business objectives, comprehension of the As-Is scenario and defining the goals of the analyses.

Data Understanding: In this phase, the focus is to collecting and familiarizing with the data. It involves tasks such as initial data collection, data description, data exploration, and data quality verification.

Data Preparation: Data preparation involves all the tasks related to constructing the final dataset that will be used for modeling. This can include data transformation and cleaning, as well as feature engineering.

Modeling: In the modeling phase, various models are selected and applied to the dataset. Choosing the best model is a difficult task, since it depends on the nature of the data and the type of problem that it is trying to solve. It is essential to train and test the model with different parameters to obtain the best accuracy.

Evaluation: Once models have been built, they need to be evaluated to ensure they comply with what was previously defined. This can be done using metrics and more than one model should be tested, before choosing the best approach.

Deployment: Once a model has been evaluated, it needs to be deployed into a business environment to start providing the intended insights. As this is an academic project, this phase has not been implemented.

1.5 Document Structure

The document begins with an introduction, laying the context in which this research is positioned. It delves into the problem being addressed and the main goal of the study. The methodology followed is also explained and ends with a brief overview of how the document is structured.

Following the introduction, the literature review section presented here, the document delves into the domain of demand forecasting, introducing the principles of time series analysis. This section details some forecasting models, such as the *Seasonal Autoregressive Integrated Moving Average eXogenous* (SARIMAX) and the *Gradient Boosting Decision Tree* models. To ensure a fundamented choice of the best methods to implement, some algorithm evaluation metrics are presented.

Transitioning to the core of the dissertation, there is a full chapter with an in-depth description of the dataframe, followed by another chapter that explains the steps involved in data pre-processing and implementation steps. This core section finishes with the implementation results of the algorithms, where a comparison between them is made.

The document concludes with a reflection on the research and some ideas for future work in this domain are presented, providing some potential paths to continue this study.

Chapter 2

Literature Review

In today's fast-paced business environment, understanding future demand is crucial for any organization that wishes to stay ahead of its competitors, as it enables organizations to make informed decisions, manage resources efficiently, and take advantage of potential opportunities.

This chapter provides a comprehensive insight into the domain of demand forecasting. Although there are numerous models and methods available, the focus will be in two specific models: SARIMAX and Gradient Boosting. SARIMAX, a time series forecasting method, combines seasonal decomposition, autoregression, and moving averages to forecast demand. On the other hand, Gradient Boosting is a machine learning technique that builds forecasting models on data samples, ensuring precision and adaptability.

2.1 Introduction to Demand Forecasting

Companies performance are affected by the unpredictability of Supply Chain Management. To mitigate the impact on performance, it is necessary to analyze all business-related data. Until recently, data collected by companies was scarce. Nowadays, in the digital era, the challenge focus on how to organize, prioritize and define a meaning/pattern for the data collected from all sources, i.e., there is a vast amount of available data and, to take advantage of it, it is necessary to know how to correlate them and give them a purpose.

In the demand forecasting domain, data is a key resource for enhancing prediction accuracy. These forecasts are crucial, serving as tools for future planning and promoting informed decision-making (Armstrong 1988), which leads to performance optimization, costs reduction and sales & profits increase. Kocaoglu, Acar, and Yılmaz (2014) defines demand forecasting as the process of analyzing and regulating information that enables forecasts of future sales, ensuring that resources are allocated efficiently and customer satisfaction remains high.

Numerous studies suggest that the future often reflects the past. Thus, to make accurate predictions about the future, the past must be analyzed (Kocaoglu, Acar, and Yılmaz 2014). Many factors influence customer demand, such as lead time of product replenishment, advertising or marketing efforts, price discounts, the state of the economy and actions that competitors have taken. In order to predict demand, a company must identify the relationship between those factors and past demand (Chopra and Meindl 2012).

A good demand forecasting should be a crucial task for companies, since it avoids over/under stocking and missed sales opportunities. However, an accurate demand forecast is a challenge for professionals. While a few years ago companies used traditional and very limited forecasting models to predict demand, with the current amount of data available in all the companies' systems, it has been necessary to develop new types of models (Babaee Tirkolaee et al. 2021).

Researches show that in the last decade, several companies have disappeared due to a misinterpretation of market signs and their inability to adapt to the swift technological advancements and rapid growth in consumer demands and expectations. In today's digital era, where data are continuously generated, with the power of modern computing, it is possible to process large amounts of data, making real-time decision making a reality. This can be assured using machine learning algorithms (Aamer, Yani, and Priyatna 2021). This technology, together with the data sources currently available, provide the capability to foresee market changes, making it possible for companies to react and adapt quickly to these changes. For sectors that experience high variability and sensitivity to external factors, such as the bike-sharing industry, this evolution is essential.

In the following sections, a deeper dive into demand forecasting methods will be done, focusing on two different models.

2.2 Time Series Analysis

A time series is a sequence of chronologically ordered values, recorded at a specific time. Time series forecasting aims to predict future values based on historical data. It also helps to understand patterns, such as trends, seasonality and cycles. By understanding historical patterns and accurate forecasting, businesses and organizations can make informed decisions, leading to better business (Hyndman and Athanasopoulos 2021).

These are the main components of a time series (Hyndman and Athanasopoulos 2021), (Torres et al. 2020):

- **Trend**: The underlying direction in which the data moves over time, increasing or decreasing. When a trend changes from an increasing to a decreasing graph, it is called a changing direction.

Some types of trend are linear, exponential or parabolic trend. In the linear trend, data increases or decreases at a constant rate, where the function represents a straight line. Exponential trend differs from linear trend in the growth rate, since it represents an increasing or decreasing rate rather than a fixed number. The function is represented by a curve. Parabolic trends are characterized by the shape of a parabola (U), in which the rate of change starts slowly, accelerates significantly, and then decelerates again, approaching a turning point where it either stabilizes or reverses.

- **Seasonality**: Repetitive patterns in data that occur at regular and known intervals, affected by seasonal factors such as events or festivities, climate periods or even the day of the week or month of the year. Understanding seasonality is imperative for businesses to allocate resources, adjust marketing strategies, and anticipate demand fluctuations.

- **Cyclic**: Cyclic patterns represent long-term variations in the dataset, where data are displayed in an ascending or descending order without a fixed frequency, so they do not have the predictability of seasonality. They can vary in length and usually arise from economic situations.

- **Noise**: The random variation in data that does not fall under trend, seasonality or cyclical components, representing the inherent unpredictability of a dataset. This noise can result from data collection errors or from genuine random variations.

Figure 2 illustrates each of the aforementioned patterns.

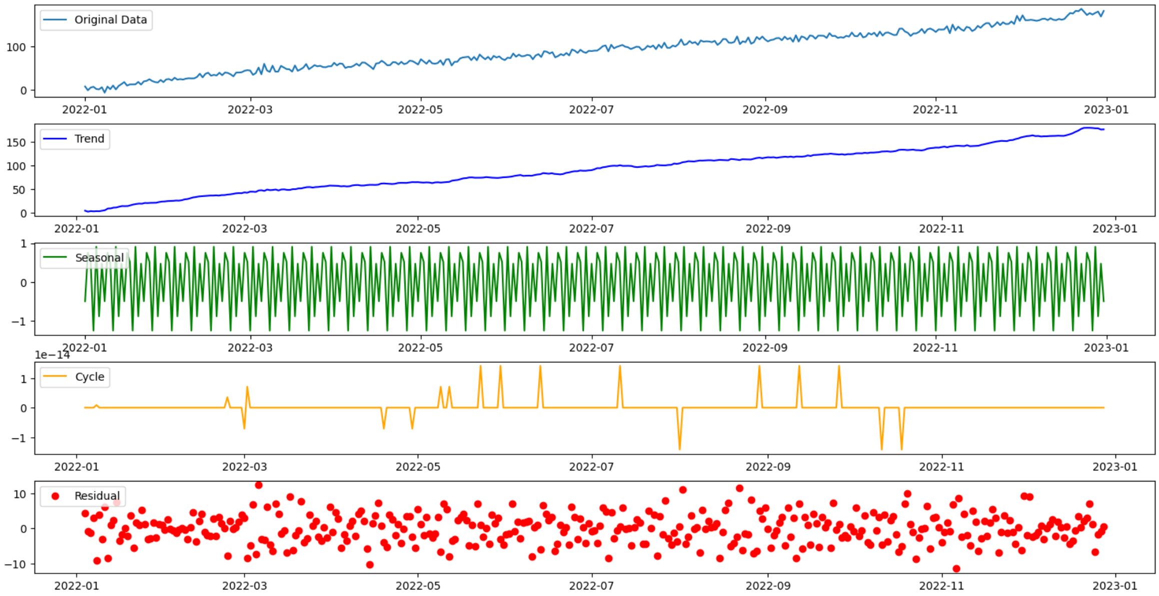


Figure 2. Decomposition of Time Series into Trend, Seasonality, Cycle and Noise

Although traditional models have been effective and reliable for many applications, the massive flow of data and the complexity of modern problems have stimulated the development of more advanced models. The following sections will delve a little deeper into each of these approaches.

2.3 Forecasting Models

There are numerous techniques for time series forecasting, some of which are more traditional and others are more advanced. Traditional time series forecasting models, which have been in use for decades, are based on statistical methods and understanding of historical patterns. These traditional models include Autoregressive Integrated Moving Average (ARIMA), Exponential Smoothing State Space Model (ETS), and Holt-Winters method, to name a few. They often work best when data has a clear trend or seasonal patterns.

On the other hand, more advanced forecasting techniques use the power of modern computational capabilities and the vast amounts of data currently available. These methods include machine learning algorithms, such as Neural Networks or Decision Trees, which can capture complex relationships and patterns in the data.

Although there are a vast number of techniques to predict demand, this section will focus on the following two approaches: SARIMAX and *Gradient Boosting*. SARIMAX is an extension of ARIMA, which is a very popular model as it is highly effective for forecasting problems. In addition, SARIMAX complements ARIMA by incorporating seasonality and external variables to the model, factors that allow it to improve the predictive capacity of the model. *Gradient Boosting* is a decision tree based model that helps discover complex patterns in the data that simplest models may not capture. In addition, it allows the process of large data volumes without requiring large computational capacity.

2.3.1 Seasonal AutoRegressive Integrated Moving Average with eXogenous Regressors Model

SARIMAX is an extension of ARIMA that incorporates seasonal patterns and exogenous variables, allowing for a more comprehensive time series forecast. To understand SARIMAX, it is important to first be familiar with ARIMA.

ARIMA stands for AutoRegressive Integrated Moving Average and is one of the most versatile and widely-used time series forecasting methods (Hyndman and Athanasopoulos 2021). It combines three components: AutoRegressive, Integrated, and Moving Average. Each component addresses a different property of the time series:

- **AutoRegressive (AR):** Uses the relationship between an observation and a number of lagged observations (previous periods);

- **Integrated (I):** Represents the number of differences needed to make the series stationary, meaning that its statistical properties do not change over time;

- **Moving Average (MA):** Uses the dependence between an observation and a residual error from a moving average model applied to lagged observations.

Before deep dive into each of the concepts, it is essential to understand the notion of stationary and its significance in ARIMA forecasting.

Stationarity refers to a property of a time series in which its statistical characteristics, such as mean, variance, and autocorrelation, remain consistent over time. This means that the series does not exhibit trends, seasonality or cycles (Hyndman and Athanasopoulos 2021). When a time series is stationary, its patterns and structures are easier to discern, leading to more accurate and insightful models. The presence of trend, seasonality and cycle makes the modeling process more complex and potentially less accurate, as it is trying to predict changes in patterns.

Before starting the modeling process, it is imperative to transform a non-stationary time series into a stationary one, a procedure known as differencing. Differencing helps to stabilize the time series by removing the difference in the level of a time series. This difference is calculated by subtracting the previous observation from the current observation. This step eliminates or reduces the impact of trends and seasonality. Equation (2.1) illustrates the differencing procedure.

$$d_t = y_t - y_{t-1} \tag{2.1}$$

d_t	Difference of the series at time t
y_t	Value of the series at time t
y_{t-1}	Value of the series at time $t-1$

Once stationarity is achieved in the data, the forecasting model can be developed.

In an autoregressive model, the variable of interest is predicted through a linear combination of its past values, which means a regression of the variable against its own historical data. The moving average model uses past forecast errors,instead of utilizing past values of the prediction variable.

By combining differencing, autoregression, and moving average, a non-seasonal ARIMA model is created. This model can be written using Equation (2.2).

$$y'_t = c + \phi_1 y'_{t-1} + \cdots + \phi_p y'_{t-p} + \theta_1 \varepsilon_{t-1} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \tag{2.2}$$

y'_t	Difference of the series at time t
c	Constant term
$\phi_1, \phi_2, \ldots, \phi_p$	Parameters of the autoregressive part
$y'_{t-1}, y'_{t-2}, \ldots, y'_{t-p}$	Lagged (past) values of the differenced series
$\theta_1, \theta_2, \ldots, \theta_q$	Parameters of the moving average part
$\varepsilon_{t-1}, \varepsilon_{t-2}, \ldots, \varepsilon_{t-q}$	Lagged (past) values of the forecast errors
ε_t	Forecast error at time t

The autoregressive component reflects the influence of the past p values of the differenced series on the current differenced value. The moving average part captures the influence of past errors, denoted as ε, on the current value. The constants, represented by ϕ and θ, represent the weight of these past terms.

In summary, the ARIMA model combines the AR and MA components, while accounting for differencing to handle the non-stationarity of time series data. The aim is to capture patterns in historical data to make future predictions. The values of ϕ and θ are typically defined based on statistical criteria.

Although ARIMA can handle many time series data, it struggles with those that exhibit seasonal patterns. To deal with these patterns, SARIMA models are indicated. SARIMA aims to seasonal-ARIMA and adds a seasonal component to the ARIMA model. This makes it suitable for forecasting the values of a series that exhibit seasonal patterns.

To further enhance the understanding of seasonality, SARIMA uses a concept called "backshift" notation. The backshift operator shifts the time series data backward a seasonal period, that could be, for example, a week, a month or a trimester, allowing the model to consider previous observations when making predictions. By incorporating backshift notation, SARIMA efficiently captures the repeated seasonal variations in the data, ensuring more accurate forecasts especially for series that oscillate with a certain periodicity.

Additionally, the *X* in SARIMAX stands for exogenous variables. SARIMA models focus on the past values of the variable being predicted. However, in certain situations, external variables are essential to improve the model's performance. Exogenous variables are factors that are not inherently part of the time series data being forecast, but they might influence its behavior. By incorporating these variables, SARIMAX uses correlation analysis to take outside influences into account, offering a more comprehensive view of the factors affecting the series and potentially leading to more accurate forecasting results (Elshewey et al. 2023).

2.3.2 Gradient Boosting Decision Tree

Gradient Boosting, originally created for classification models, is an ensemble learning method that is widely employed in regression models (Saupin 2022). It uses a set of decision trees that learn sequentially, improving the performance of the previous model. It is widely used because it does not require a very powerful computing capacities while dealing with large amounts of data.

Decision trees work by breaking down a complex decision-making process into a series of simpler decisions. Starting from the root node, the data is divided based on certain conditions. Each node represents a test on a specific feature, each branch represents an outcome of that test, and each leaf node represents a prediction (Brownlee 2016).

The objective of the Gradient Boosting technique is to convert weak predictors into strong predictors, using weak learners and a loss function. The algorithm seeks to find an additive model that minimizes the loss function. The gradient (e.g., residual) is calculated, and a model is then fitted to the residuals to minimize the loss function. The current model is added to the previous model, and the procedure continues for a number of iterations specified by the user (Kuhn and Johnson 2018).

Gradient Boosting involves three elements (Kuhn and Johnson 2018):

Loss Function: A loss function is a mathematical function that quantifies how well a predictive model's predictions match the actual values. In gradient boosting, when dealing with regression problems, squared error is frequently used, whereas classification tasks might leverage logarithmic loss.

Weak Learner: In gradient boosting modeling, decision trees predominantly serves as weak learners. These trees facilitate the process of producing real-value outcomes based on their splits. While constructing these trees, a methodical approach is adopted. The optimal split points are identified based on some criteria, ensuring that the ultimate goal of loss minimization is achieved.

Additive Model: Trees are added sequentially to correct the error made by the sum of the preceding trees. For that, a gradient descent procedure is used to minimize the loss when adding trees.

Figure 3 shows the architecture of this model. Starting with an initial dataset, data is fed into the first decision tree which serves as a weak classifier. After the tree makes its predictions, the prediction residuals (differences between the predicted and actual values) are calculated. These residuals inform the weighting of data points in the subsequent tree. This iterative process continues, with the predictions of each tree influencing the data weighting for the next tree. The predictions from all these weak classifiers are combined to form a single, more powerful ensemble - the strong classifier (Kuhn and Johnson 2018; Deng et al. 2021).

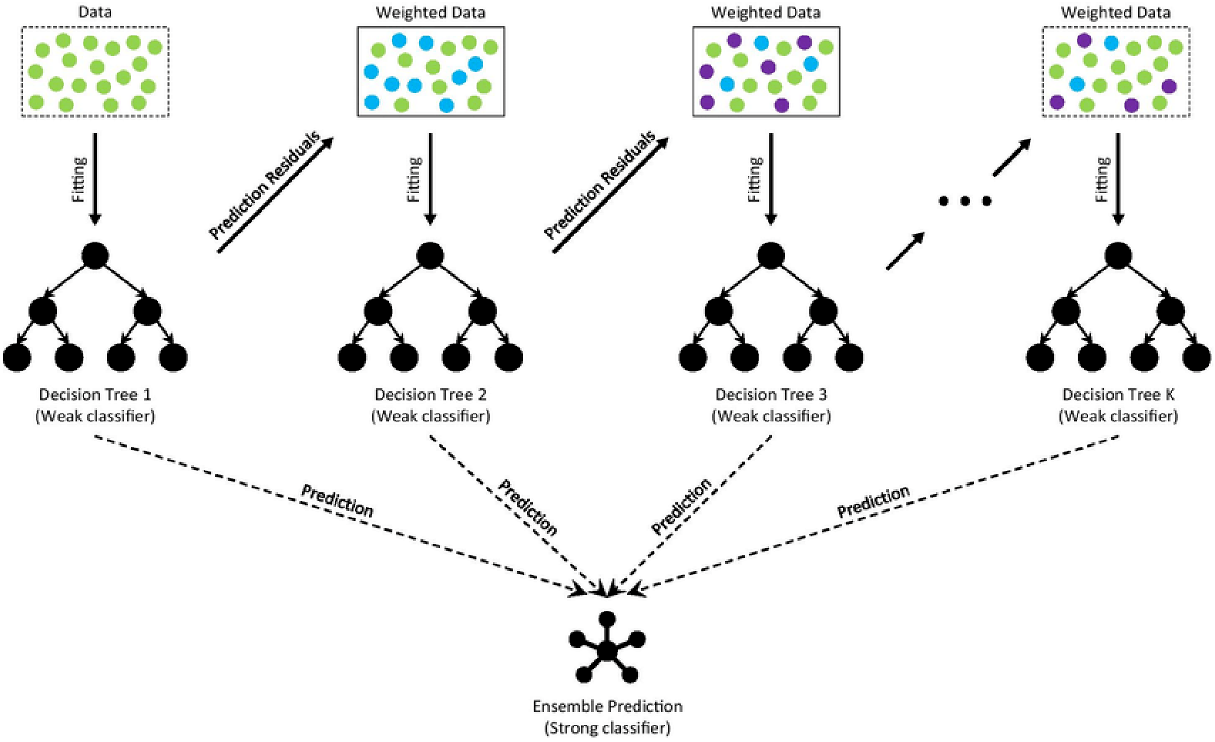


Figure 3. Gradient Boosting Architecture ((Deng et al. 2021)

Once the model is trained, it provides insights into which features are most influential in making predictions, allowing practitioners to focus on critical variables and potentially speed up data collection processes. Incorporating features that have a limited impact on the predicted variable might not only increase computational demands, as it turns the model more complex, but also compromise the model's efficacy. Additionally, although Gradient Boosting can work effectively without hyperparameter tuning, when properly tuned, it can achieve even higher accuracy.

2.4 Demand Forecasting Model's Evaluation

In the domain of data analytics, understanding the accuracy and reliability of a model's predictions is critical. To this end, various error metrics were developed to evaluate the performance of models in various situations. These metrics serve as benchmarks, providing insights into how close a model's predictions align with actual outcomes and highlighting potential areas of improvement.

Some of that metrics are (Badulescu, Hameri, and Cheikhrouhou 2021; Barrera-Animas et al. 2022):

Mean Absolute Error, MAE:

$$MAE = \frac{1}{n}\sum_{i=1}^{n}|y_i - \hat{y}_i| \tag{2.3}$$

Quantifies the average of the absolute difference between predictions and actual values. It is not particularly sensitive to large outliers, since it does not square the errors. This makes it a more robust metric when dealing with data that might contain massive unexpected spikes or drops. It is on the same scale as data.

Mean Squared Error, MSE:

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2 \tag{2.4}$$

Similar to MAE, but squares the result of the difference. It measures the average of the squares of the errors between the values predicted by the model and the actual observed values. It can be influenced by outliers in the data and, since the errors are squared, it assigns greater weight to larger errors. This means that larger errors have a proportionally greater impact on the final result.

Root Mean Squared Error, RMSE:

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2} \tag{2.5}$$

Is the square root of the Mean Squared Error, MSE. It converts the error metric back to the same unit as the original data. This makes RMSE more interpretable than MSE, giving a sense of the average magnitude of the error. It is sensitive to outliers.

Root Mean Squared Logarithmic Error, RMSLE:

$$\text{RMSLE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(\text{log}(p_i + 1) - \text{log}(a_i + 1))^2} \tag{2.6}$$

Similar to RMSE, but it calculates the logarithmic difference between the model results and the observations, reducing the influence of a large error when the rate of observations is higher than the model results.

Mean Absolute Percentage Error, MAPE:

$$MAPE = \frac{100\%}{n}\sum_{i=1}^{n}\left|\frac{y_i - \hat{y}_i}{y_i}\right| \tag{2.7}$$

Expresses the absolute average error in terms of percentage. It is sensitive to outliers.

R-squared, $\mathbb{R}^2$:

$$\mathbb{R}^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2} \tag{2.8}$$

Provides an indication of model suitability. It ranges from 0 to 1, with higher values indicating a better fit to the data. R-squared represents the proportion of variance in the dependent variable explained by the independent variables in the model.

Chapter 3

Database Description

In this chapter, a deep dive at the data was conducted, with the aim of obtaining a holistic comprehension of the rentals.

3.1 Building the Database

The London bike-sharing system that is analyzed in this dissertation refers to the ”Santander Cycles” as it is sponsored by Santander. It was launched in 2010 and has become an essential part of London’s transportation system. The bikes are available for rent at stations spread across various points in the city, and users can take the trips they need, as long as they leave the bikes at one of these stations. There are multiple rates, with people paying per individual trip (£1.65 for every 30 minutes) or opt for a monthly fee (£20) or an annual fee (£120). Payments can be made at the stations or via app. If bikes are not returned or are damaged, a charge of up to £300 will be applied. The system was created as part of an initiative to promote greener means of transport, reduce congestion, and encourage more people to cycle, whether for work, leisure, or tourism. More recently, e-bikes have been introduced into this rental system (Transport for London 2023).

The original database, supplied by the governmental entity *Transports for London* , contained only data on the bike rentals, specifically the bike ID, the duration of the trip, the initial and final stations of the trip, the day and time it started and ended, so there was a record for each rental that occurred.

To understand the impact that other external variables would have on people’s predisposition to rent a bike, three more databases were added to the original, one with meteorological data, with daily data on the weather conditions, such as temperature, cloud cover and precipitation, another with data about fuel prices, with daily data on gasoline and diesel fuel, and lastly, a database that contained the public holidays and festive days celebrated in the city of London.

The database with meteorological data was obtained from the *European Climate Assessment & Dataset*

(ECA&D 2023), an entity responsible for collecting climatic data from all over Europe. The weather station used in this project is the one located near Heathrow Airport. The data on fuel prices were extracted from the UK government website (UK Government 2023), on a tab that provides statistics from various governmental departments. The database of public holidays and festive days was compiled based on information found on online sites, including government portals and websites promoting the city of London.

Before delving into the analyses and model implementations, it is essential to understand the structure and insights of the dataset in use. For clarity, a comprehensive table has been created, showcasing each variable contained in the dataset (Table 1). This table lists the variable names, their description, their datatype and variable type.

Table 1: Description of variables

Variable Type	**Variable Name**	**Datatype**	**Data Description**
Numeric Ratio	Pump Price in £/Litre ULSP	float	Price per litre of gasoline fuel
Numeric Ratio	Pump Price in Pence/Litre ULSD	float	Price per litre of diesel fuel
Categorical Nominal	Holiday	str	Description of the commemorative date
Categorical Binary Symmetric	is Holiday	bool	If it is a Holiday: True; Otherwise: False
Numeric Ratio	Cloud Cover	float	Cloud cover measurement in oktas
Numeric Ratio	Sunshine	float	Sunshine measurement in hours
Numeric Ratio	Global Radiation	float	Irradiance measurement in Watt per square meter (W/m2)
Numeric Interval	Max Temp	float	Maximum temperature recorded in degrees Celsius (°C)
Numeric Interval	Mean Temp	float	Mean temperature in degrees Celsius (°C)
Numeric Interval	Min Temp	float	Minimum temperature recorded in degrees Celsius (°C)
Numeric Ratio	Precipitation	float	Precipitation measurement in millimeters (mm)
Numeric Ratio	Pressure	float	Pressure measurement in Pascals (Pa)
Numeric Ratio	Snow Depth	float	Snow depth measurement in centimeters (cm)
Numeric Ratio	Rental ID	int	ID of the rental
Numeric Ratio	Duration	float	Duration of the rental (seconds)
Numeric Ratio	Bike ID	int	ID of the bike used
Numeric Ratio	Start Station ID	int	ID of the Start Station
Categorical Nominal	Start Station Name	str	Name of the Start Station
Numeric Ratio	End Station ID	int	ID of the End Station
Categorical Nominal	End Station Name	str	Name of the End Station
Numeric Ratio	Start Day	datetime	Day the rental started
Numeric Interval	Start Hour	str	Hour the rental started
Numeric Ratio	End Day	datetime	Day the rental ended
Numeric Interval	End Hour	str	Hour the rental ended
Numeric Interval	Year	int	Year of the rental
Categorical Ordinal	Month	int	Number of the month of the rental
Categorical Ordinal	Week	int	Week of the rental
Categorical Ordinal	Day	int	Day of the rental
Categorical Ordinal	Day of Week	int	Day of the week of the rental (Monday=0, Sunday=6)
Categorical Binary Symmetric	is Weekend	bool	If it is the weekend: True; Otherwise: False

3.2 Rental patterns

To better understand rental patterns, it is fundamental to visualize the temporal variations of the rentals.

Figure 4 and Figure 5 provide information on the start and end times of rentals, categorized by each day of the week.

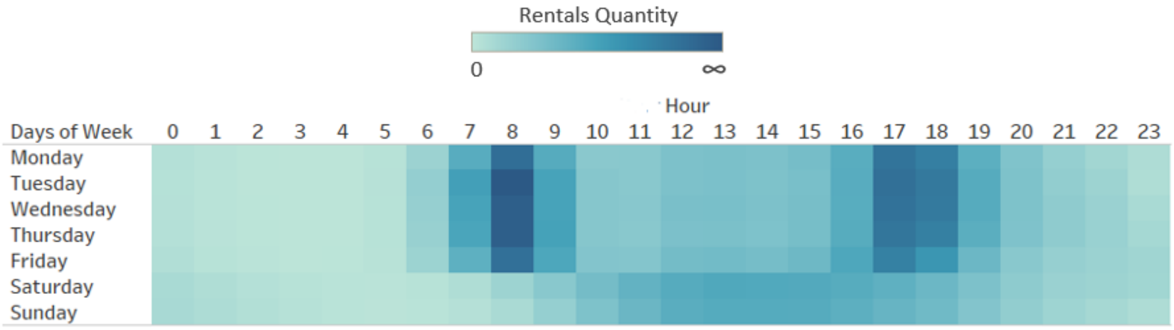


Figure 4. Bike rental frequency by start hour and day of the week

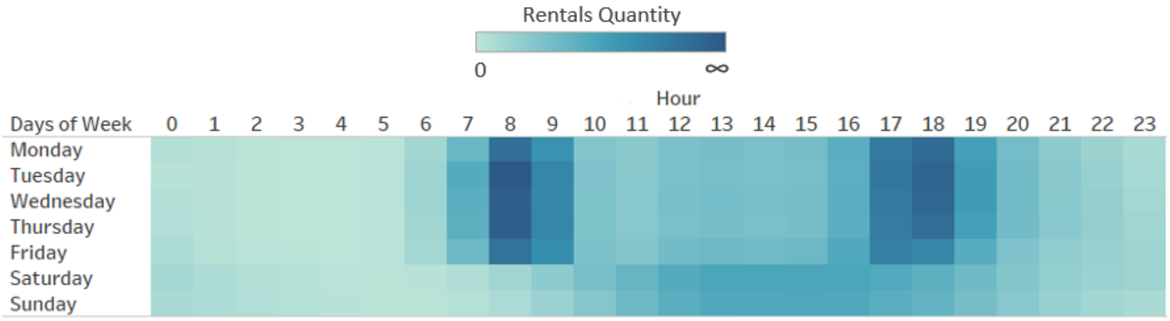


Figure 5. Bike rental frequency by end hour and day of the week

As it is possible to observe, the peak hours during weekdays, are in the morning, between 6am and 9am, and again in the late afternoon between 4pm and 7pm, which aligns closely with the typical work and school schedules, reflecting the times when people are start and finish their daily commitments. In contrast, the weekend rental data reveals a less pronounced peak in activity during the 10am to 6pm period, which represents people's leisure time. Furthermore, in tourist destinations, weekends often witness more visitors who use bikes to explore the area. This can contribute to an increase in bike rental activity during the daytime hours of the weekend.

Comparing Figure 4 and Figure 5, a subtle shift to the right during peak times is observed, corresponding to the start of rentals, in addition to the length of the trip.

Figure 6 shows the dispersion of the rentals on each day of the week. Midweek exhibits the highest number of rentals, while there is a decreasing trend in rental activity as the weekend approaches. The weekend days record the lowest number of rentals, which can reinforce the use of this type of transport for commuting between transports or going to work.

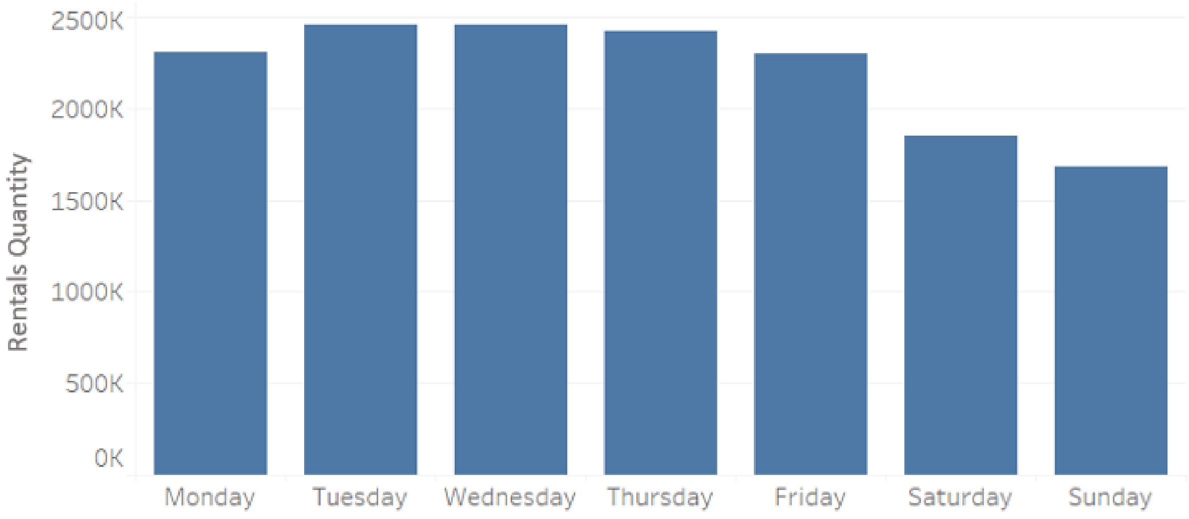


Figure 6. Bike rental frequency by day of the week

Figure 7 demonstrates that, as the seasons change throughout the year, so do the patterns of bike rentals in London.

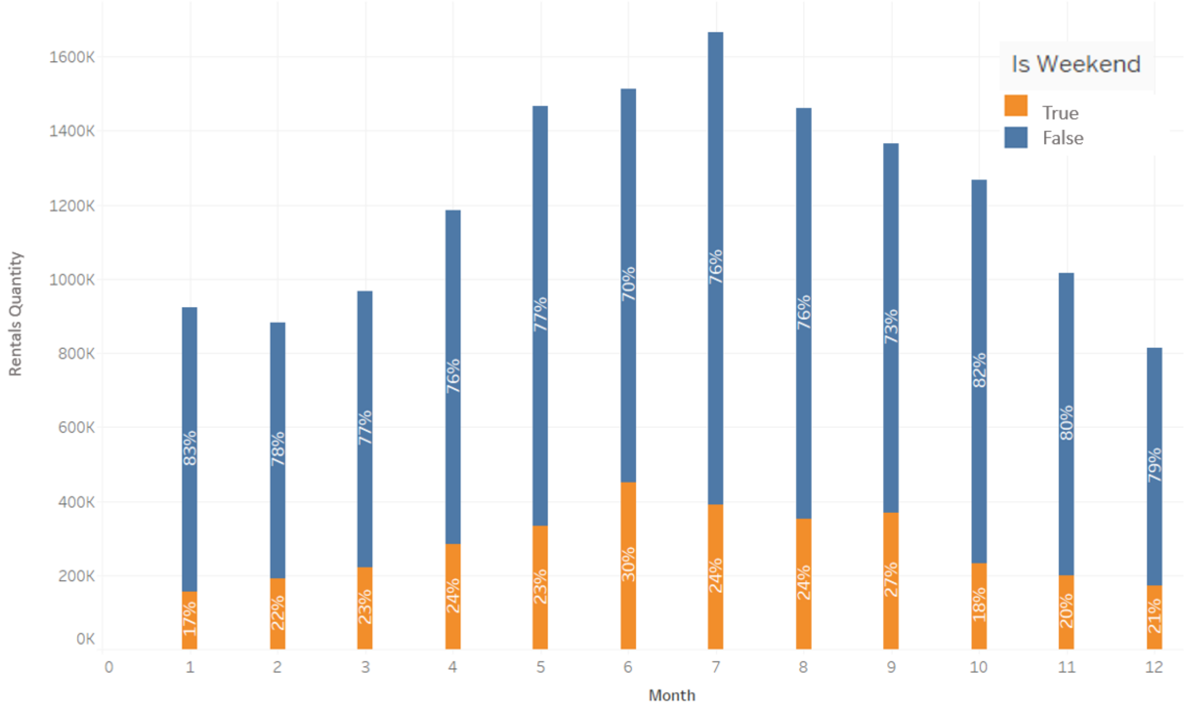


Figure 7. Bike rental frequency by month. 1-January, 12-December

The hotter months, from May to August, emerge as the peak bike rental season. During this period, with warm temperatures and longer daylight hours, many people take advantage of the good weather to use bikes as a means of transportation or just to enjoy leisurely rides.

However, as the calendar turns to the fall and winter months, there is a noticeable decrease in the frequency of bike rentals. The transition to colder and often bad weather prompts fewer individuals to opt for bike as a mean of transportation.

Comparing the blue and orange columns, it is possible to observe that the trend described above applies to both weekdays and weekends.

Figure 8 shows the daily bike rentals divided by month for the year 2018. It clearly illustrates the impact that some holidays and festive days have on bike rentals, which could suggest that these days may be correlated with rentals. This influence can be manifested by an increase in the number of rentals (red arrows) or a decrease (black arrows). On one hand, there are holidays that attract people to outdoor leisure and recreation activities. On the other hand, there are holidays dedicated to more specific activities, such as staying home, attending religious events or visiting family and friends.

The blue brackets in Figure 8 correspond to weekends. A closer look reveals that, typically, fewer bikes are rented on weekends, as previously shown.

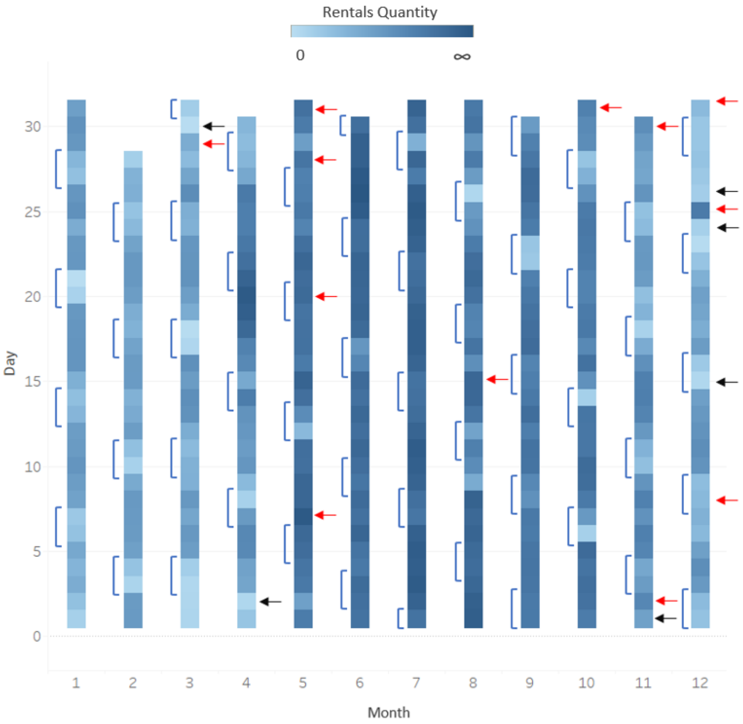


Figure 8. Bike rental daily frequency by month in 2018

Figure 9 illustrates all daily bike rentals throughout 2018. Through its analysis, it becomes evident that there is a seasonality regarding the months of the year and the days of the week, with an upward trend as the warmer months approach, reaching the peak during summer, and a downward trend as the colder months arrive. Again,the weekends exhibit lower rental volumes, increasing on weekdays.

The orange points marked on the graph represent weekends. It is notable that the majority of the lower peaks correspond to weekends, while the upper peaks represent weekdays.

In Figure 9 it is not possible to assess the overall trend of bike rental, as there is not enough data (there is only data from 2018 and 2019, which prevents drawing conclusions in this regard).

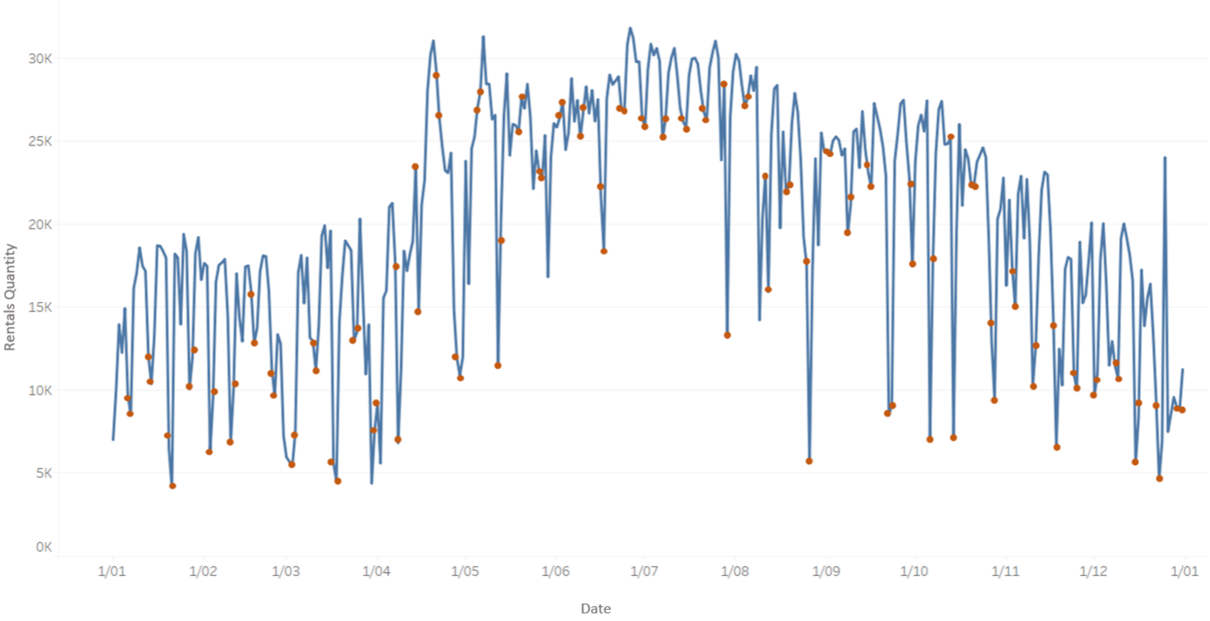


Figure 9. Evolution of bike rental daily frequency in 2018

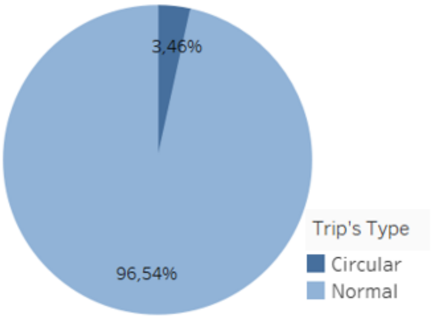


Figure 10. Type of Trips

Regarding the type of trips made, it is interesting to understand the distribution between cyclical trips, that is, trips that start and end in the same station, and trips that start in one place and end in another. Looking at the graph at Figure 10, the percentage of people who end their trip in a different location than the starting point is clear.

3.3 Bike stations

Each station also has its usage patterns. This next section addresses some of them.

Figure 11 shows the location of all the bike stations in London city, where the larger the dots, the higher the station' affluence.

From its visual representation, it is evident that the most used stations are located in the central areas of the city. In contrast, the smaller points, which represent fewer rentals, are mainly found on the periphery of the city.

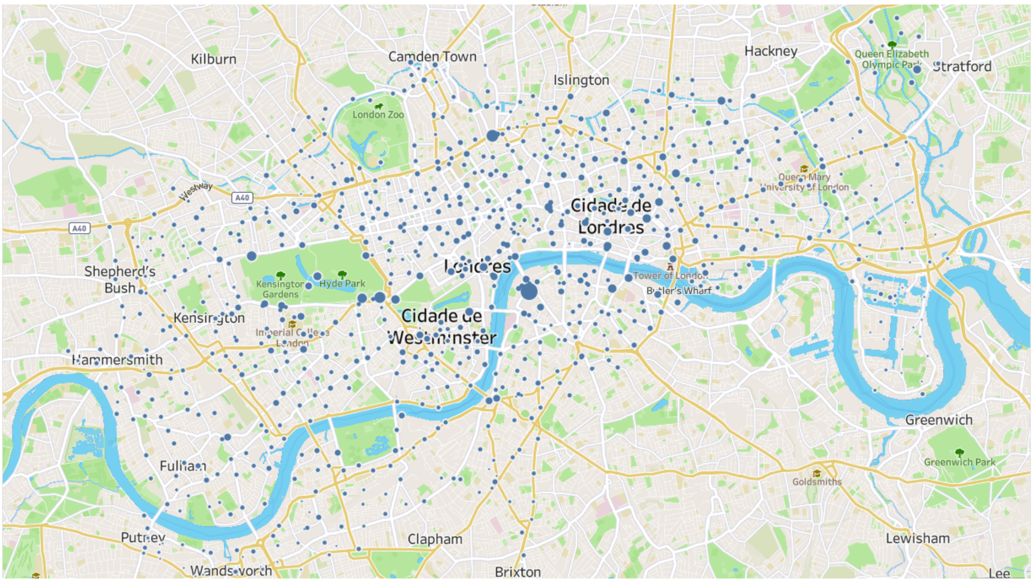


Figure 11. Bike Station's Distribution

Most rentals take place in central London areas, such as City of London, Hyde Park, City of Westminster, Queen Elizabeth Olympic Park, Waterloo and King's Kross. Hyde Park is a leisure spot, and the City of London and Westminster are not only very touristic places but also business districts, where bikes become a more attractive means of transportation due to the impracticality of using cars in congested zones. Waterloo and King's Kross are the main hubs in the London transportation network.

The maps in Figure 12 and Figure 13 represent bike rentals, specifically the departure stations (Figure 12) and the arrival stations (Figure 13), during the time frame of 4pm to 7pm, which corresponds to the end of the workday, on May 10th, 2018 (Thursday).

During the period from 4pm to 7pm, there was a predominance of rentals starting in the City of London and Hyde Park areas. The City of London will have many workers concluding their workday and some of them may choose to rent a bike to return home or to move to post-work activities. After a day at work, many may choose to enjoy a relaxing ride in Hyde Park, one of London's largest parks, which can explain

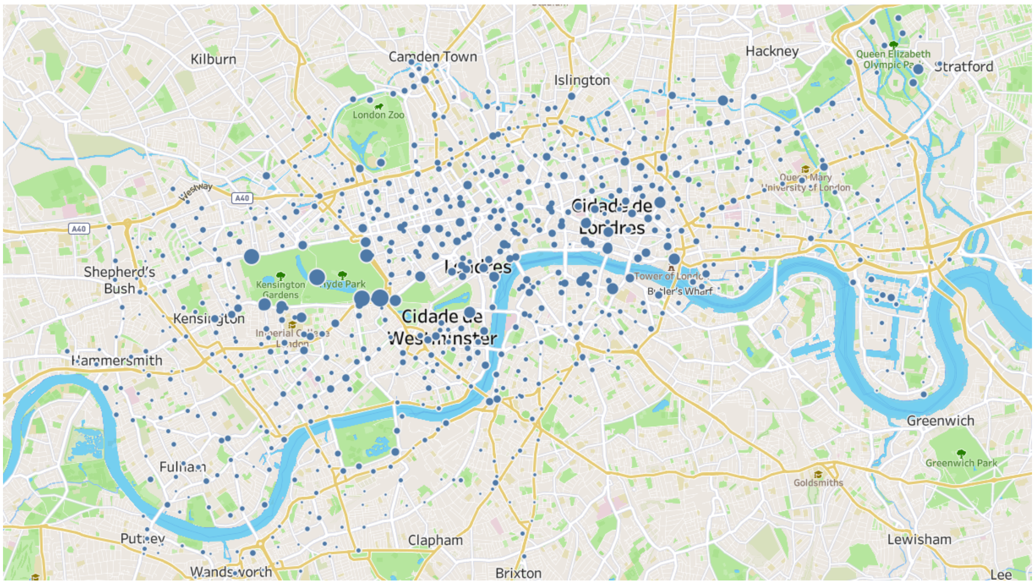


Figure 12. Bike Start Station's Distribution on May 10, from 4pm to 7pm

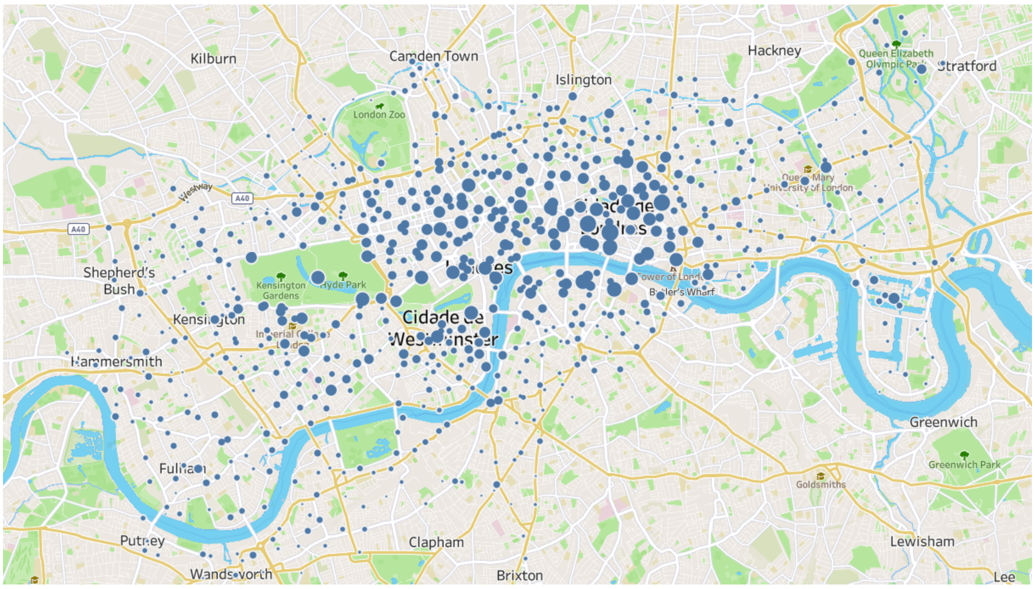


Figure 13. Bike End Station's Distribution on May 10th, from 4pm to 7pm

the high departures from this area.

On the other hand, the final stations are more uniformly distributed across the city, which may suggest that people live dispersed around the city or that they can take another type of transport from various points in the city to reach their final destination.

The chart in Figure 14 lists the most frequently used stations in descending order, with rental amounts differing between weekdays and weekends. It is clear that the most popular stations are in the areas of King’s Cross, Waterloo, and Hyde Park, as previously mentioned on the first map. At weekends, the most used stations are predominantly in the Hyde Park area, since it is a prime leisure area.

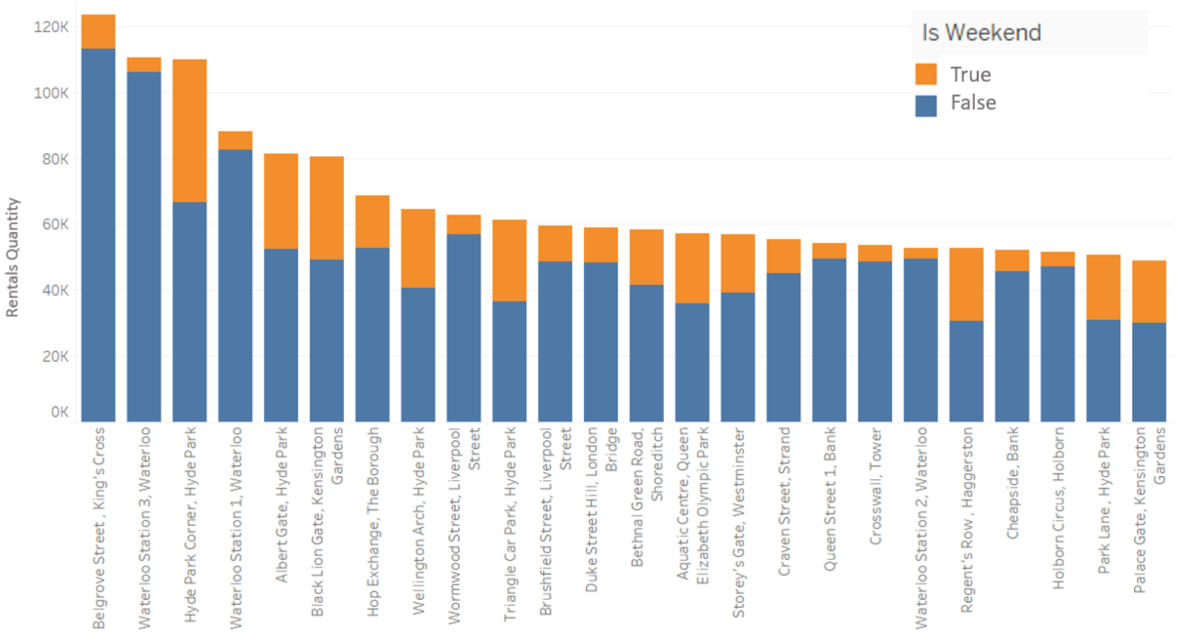


Figure 14. Most used stations

3.4 Bike Trips

The duration of bike trips (Figure 15) is predominantly less than 30 minutes, with the peak duration represented by trips of 8 minutes. Nonetheless, there are instances where the trip duration significantly exceeds the average, with some trips surpassing 24 hours (thus incurring penalties).

Table 3 shows statistical details about duration of bike trips. On average, the duration of bike trips is 19 minutes, which indicates that the majority of users tend to use the service for short, quick trips. This is further reinforced by the median value of 13 minutes, which suggests that more than half of the users rent the bikes for less than a quarter of an hour.

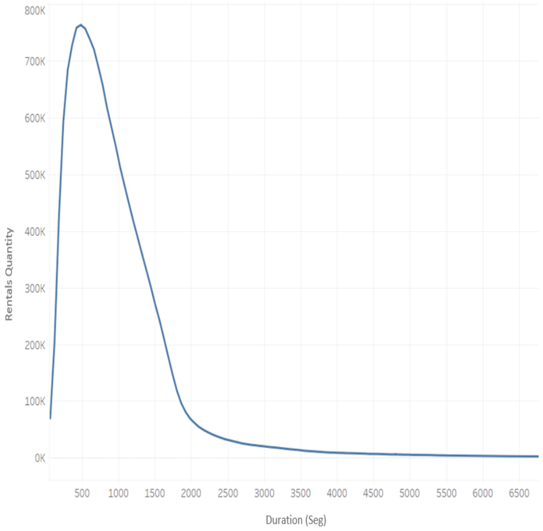


Figure 15. Bike rental's duration

Table 2. Statistical data on duration

Metric	Duration
Min	1min
Mean	19min 31seg
Max	6 Days 18H 33min
Median	13min
Standard Deviation	1H 37Seg
First Quartile	8min
Third Quartile	21min

The shortest trip is just 1 minute, which can represent cases where users may have changed their minds after renting or possibly had to deal with a bike breakdown. On the other hand, the longest rental duration is a remarkable 6 days. Such long rental periods are outliers and might be due to users forgetting to dock their bikes, potential bike thefts, or even possible system errors.

The standard deviation value of 1 hour suggests that, although a significant proportion of users remain at the mean value, there are still many who rent their bikes for a longer or shorter period of time. The outliers represented by the maximum duration are obviously affecting this value.

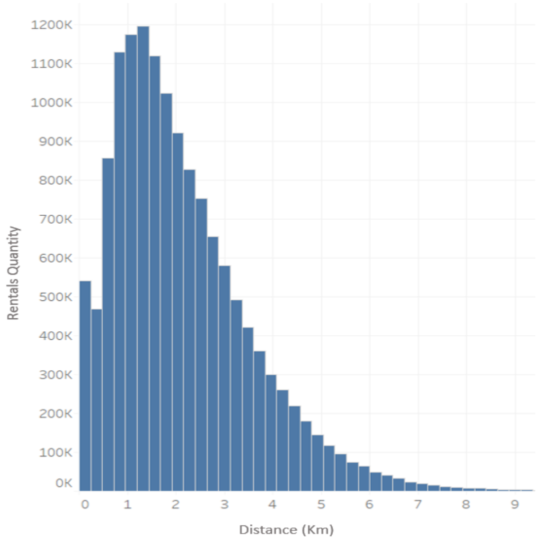


Figure 16. Bike rental's station's distance

Table 3. Statistical data on distance

Metric	Distance
Min	0km
Mean	2.1Km
Max	24Km
Median	1.8Km
Standard Deviation	1.5Km
First Quartile	1Km
Third Quartile	2.9Km

In terms of the distance traveled on each trip (Figure 16), the analyses carried out took into account the Euclidean distance between stations, and not the distance of the trip itself, since the system does not track users trips. In any case, it is possible to get an idea of the distance between the initial and final stations. The peak is reached at 1.2km. However, there are trips where the distance between the initial and final station deviate considerably from this average, with the maximum distance covered by a person being approximately 24km.

3.5 External Influences on Bike Rentals

The following figures are interpreted as follows: for each meteorological data point of x (e.g., minimum temperature of $5°C$), the average rental volume was y.

The influence of temperatures (minimum, mean and maximum) in rentals is presented in Figure 17. The behavior between the three types of temperature recorded is quite similar. Cold temperatures account for fewer rentals because they are not as suitable for bike trips. As the temperature starts to rise, there is a corresponding increase in the number of rentals.

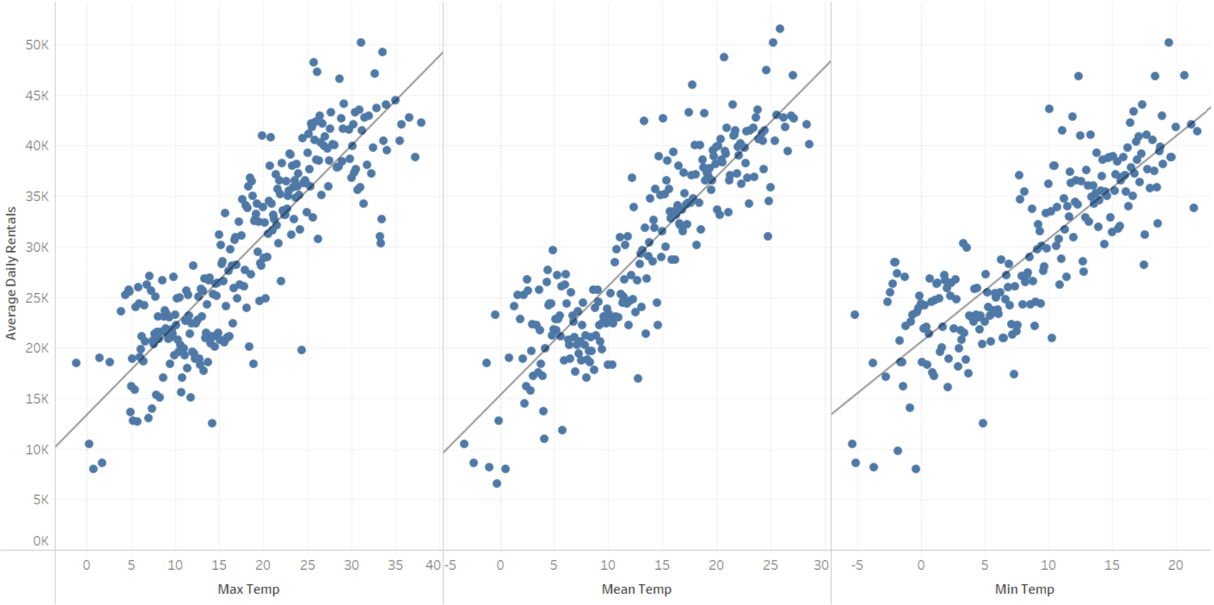


Figure 17. Temperature influence on rentals

Regarding precipitation and sunshine influence in rentals (Figure 18), there is a clear trend that days with less precipitation witness higher number of bike rentals. This can be attributed to the fact that drier conditions are more conducive for outdoor activities like biking. When there is more rain, people might be more inclined to drive a car or catch a taxi.

The relationship between the number of sunshine hours and bike rentals is the opposite (Figure 18). Nonetheless, there is some days with longer sunlight hours but fewer trips. This counter-intuitive trend may suggest that extremely long days or intense sunlight can potentially deter some users from renting bikes. Perhaps extended daylight hours, especially during the peak of summer, result in hotter conditions, making bike rental less appealing.

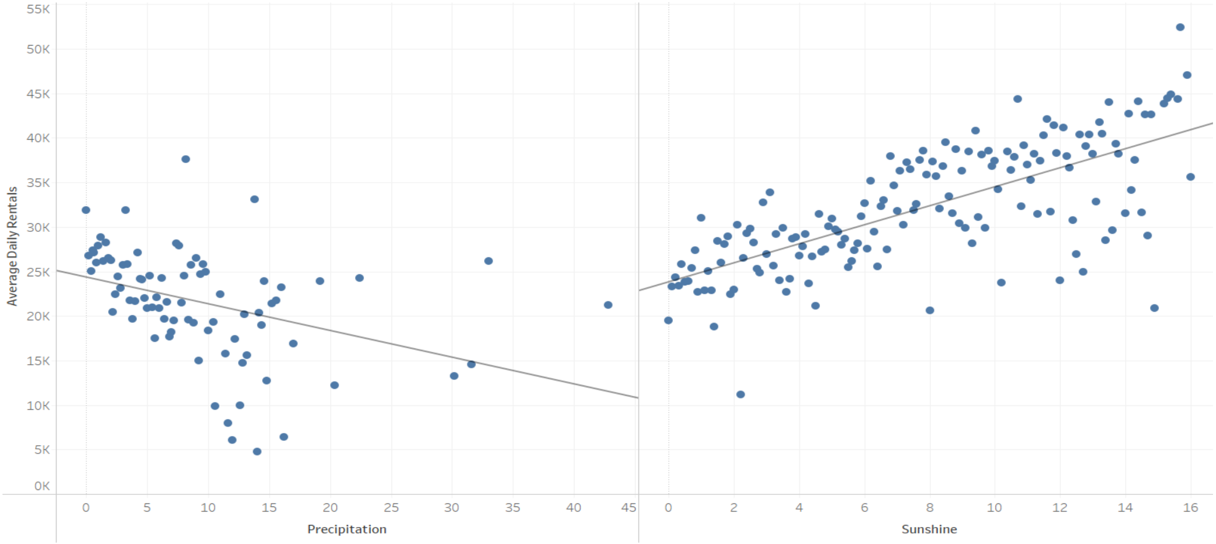


Figure 18. Precipitation and Sunshine influence in rentals

Figure 19 shows that the analysis of cloud cover measurements reveals a negative relationship with bike rentals, as there seems to be an decrease in bike rentals on days with higher cloud cover. This can be explained by the fact that, although the city is known for its predominantly cloudy weather, cloudy days often translate into rainy or foggy days, which makes cycling unpleasant.

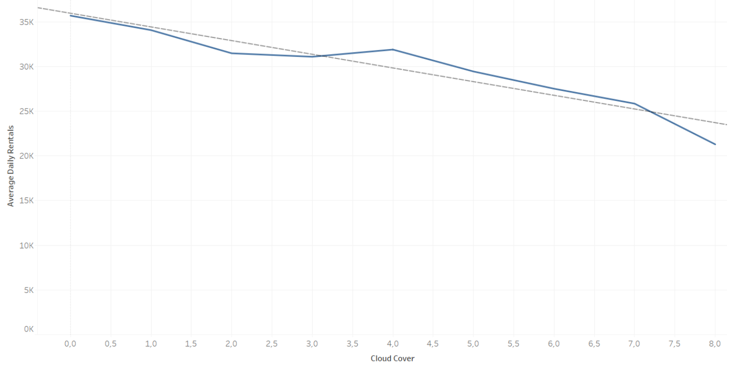


Figure 19. Cloud cover influence in rentals

The predictable analysis of the influence of fuel prices on bike rental suggests that as fuel prices rise, individuals may seek to rent bikes more often to save money in transportation. However, Figure 20 presents a slightly different narrative. This may suggest that fuel price does not influence the people's predisposition to rent a bike.

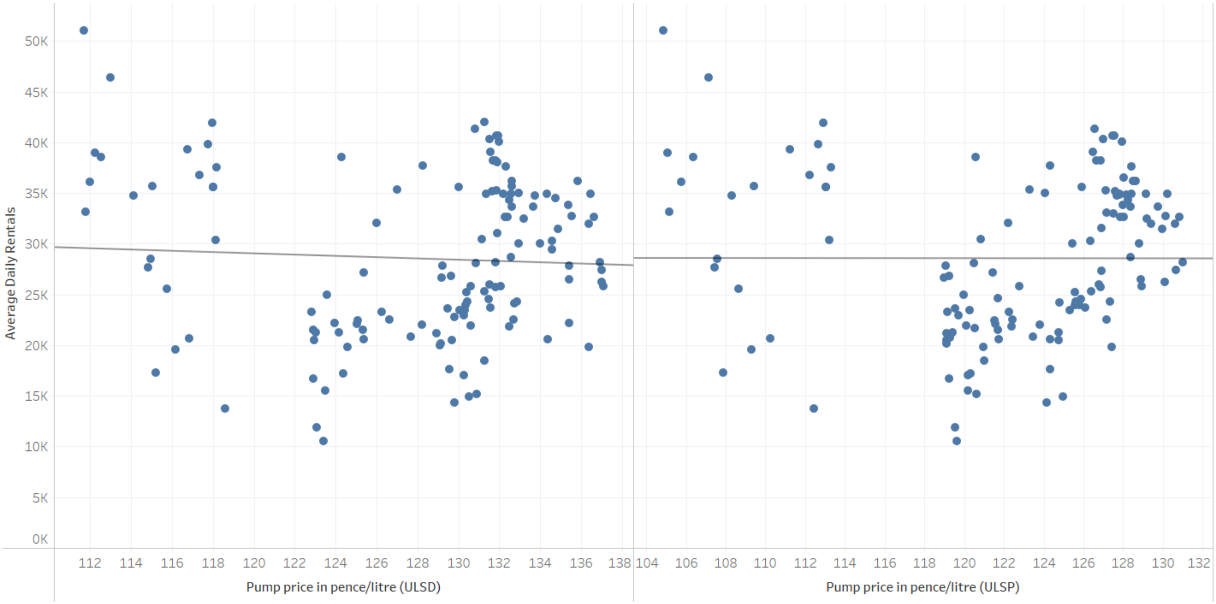


Figure 20. Fuel Price influence in rentals

To better understand the influence of each variable has on rentals, a correlation matrix is created, as shown in Figure 21, which measures the linear relationship between each pair of variables.

As it is possible to observe, the correlations can be categorized as follows:

- Weak positive correlation: Factors such as fuel price (ULSD), month, week, and holidays exhibit a marginal positive effect on rentals.

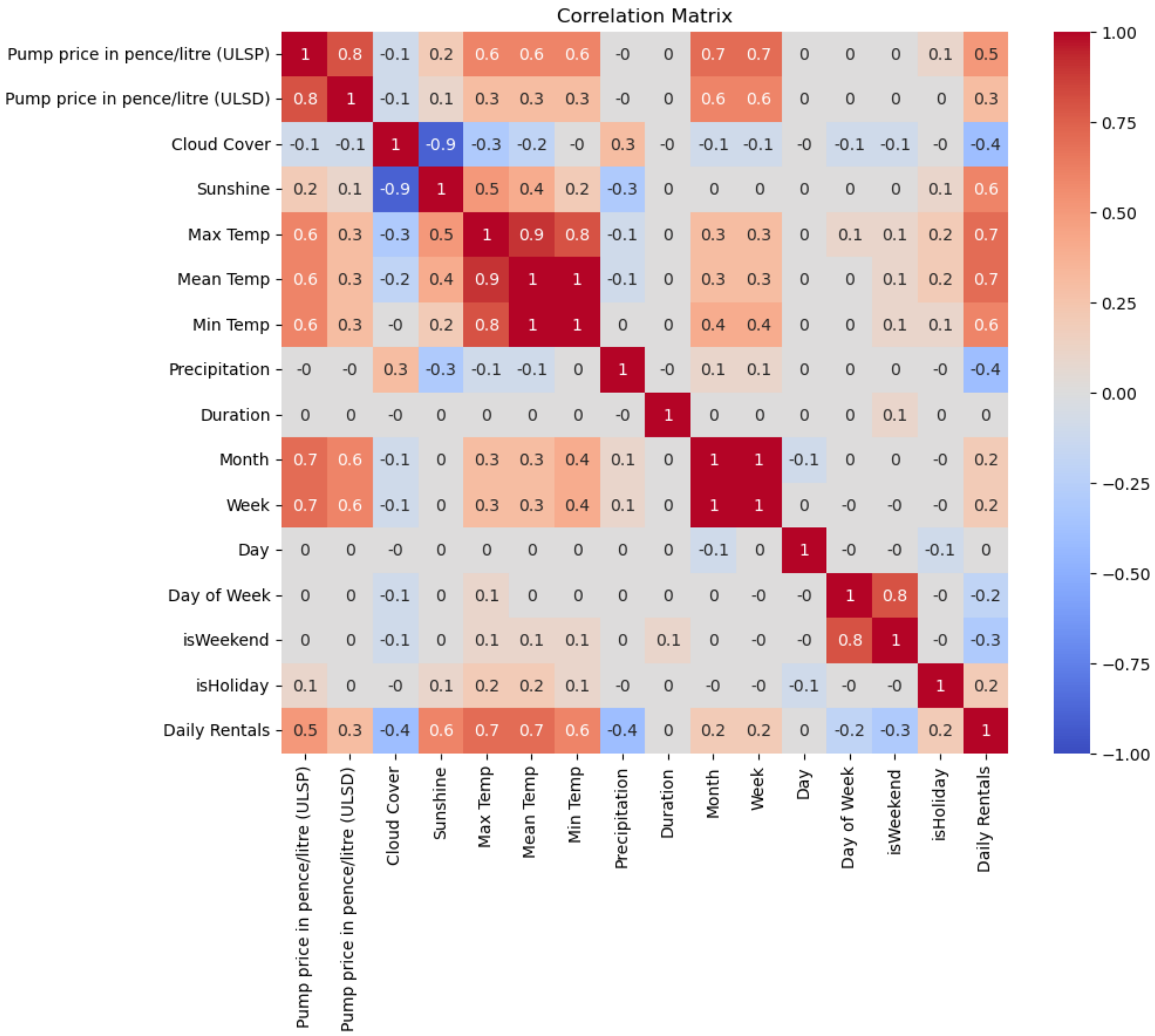
- Weak negative correlation: Day of the week and weekends show a slight inverse relationship with the number of rentals.

- Moderate positive correlation: A moderate positive correlation is observed with factors like fuel price (ULSP) and sunshine hours.

- Moderate negative correlation: Cloud cover and precipitation showcases a moderate negative correlation

- Strong positive correlation: Temperature emerges as a significant determinant, with a strong correlation with rentals.

Each of these relationships provides information on rental behaviors, which was explored individually in this previous section.



	Pump price in pence/litre (ULSP)	Pump price in pence/litre (ULSD)	Cloud Cover	Sunshine	Max Temp	Mean Temp	Min Temp	Precipitation	Duration	Month	Week	Day	Day of Week	isWeekend	isHoliday	Daily Rentals
Pump price in pence/litre (ULSP)	1	0.8	-0.1	0.2	0.6	0.6	0.6	-0	0	0.7	0.7	0	0	0	0.1	0.5
Pump price in pence/litre (ULSD)	0.8	1	-0.1	0.1	0.3	0.3	0.3	-0	0	0.6	0.6	0	0	0	0	0.3
Cloud Cover	-0.1	-0.1	1	-0.9	-0.3	-0.2	-0	0.3	-0	-0.1	-0.1	-0	-0.1	-0.1	-0	-0.4
Sunshine	0.2	0.1	-0.9	1	0.5	0.4	0.2	-0.3	0	0	0	0	0	0	0.1	0.6
Max Temp	0.6	0.3	-0.3	0.5	1	0.9	0.8	-0.1	0	0.3	0.3	0	0.1	0.1	0.2	0.7
Mean Temp	0.6	0.3	-0.2	0.4	0.9	1	1	-0.1	0	0.3	0.3	0	0	0.1	0.2	0.7
Min Temp	0.6	0.3	-0	0.2	0.8	1	1	0	0	0.4	0.4	0	0	0.1	0.1	0.6
Precipitation	-0	-0	0.3	-0.3	-0.1	-0.1	0	1	-0	0.1	0.1	0	0	0	-0	-0.4
Duration	0	0	-0	0	0	0	0	-0	1	0	0	0	0	0.1	0	0
Month	0.7	0.6	-0.1	0	0.3	0.3	0.4	0.1	0	1	1	-0.1	0	0	-0	0.2
Week	0.7	0.6	-0.1	0	0.3	0.3	0.4	0.1	0	1	1	0	-0	-0	-0	0.2
Day	0	0	-0	0	0	0	0	0	0	-0.1	0	1	-0	-0	-0.1	0
Day of Week	0	0	-0.1	0	0.1	0	0	0	0	0	-0	-0	1	0.8	-0	-0.2
isWeekend	0	0	-0.1	0	0.1	0.1	0.1	0	0.1	0	-0	-0	0.8	1	-0	-0.3
isHoliday	0.1	0	-0	0.1	0.2	0.2	0.1	-0	0	-0	-0	-0.1	-0	-0	1	0.2
Daily Rentals	0.5	0.3	-0.4	0.6	0.7	0.7	0.6	-0.4	0	0.2	0.2	0	-0.2	-0.3	0.2	1

Figure 21. Correlation Matrix

Chapter 4

Implementation

In this chapter, the methodology used to compare two distinct forecasting methods for bike rental demand is defined. It addresses in detail the data pre-processing steps and the implementation of each predictive model.

4.1 Data Pre-Processing and Preparation

The quality of the data directly influences the output, making data preparation an important step of data analytics. This section deals with the data cleaning, preparation, and splitting processes that were employed to ensure the quality of the dataset.

To optimize the analyses conducted, the datasets were segregated based on their specific scope. There were four distinct databases: one for bike rentals (with a record for each rental), another for daily weather conditions, a third for daily fuel prices, and a final one for public holidays data (with a record for each holiday or festive day). The common attribute linking all these databases is the date of the record.

4.1.1 Data Cleaning

Data cleaning is a fundamental step in the data analysis process, where missing values, duplicate values, outliers, inconsistencies and errors are addressed.

Missing Data

An examination was made on the missing values in each of the databases. During the period analysis, no missing values are found with the exception of the dataset related to weather data. This dataset has missing values concerning maximum, average, and minimum temperatures, as well as precipitation on specific days. These days were identified, and the missing values were imputed using data from the TuTiempo website, which archives historical weather information (*Climate in London - Historical Weather*

2023).

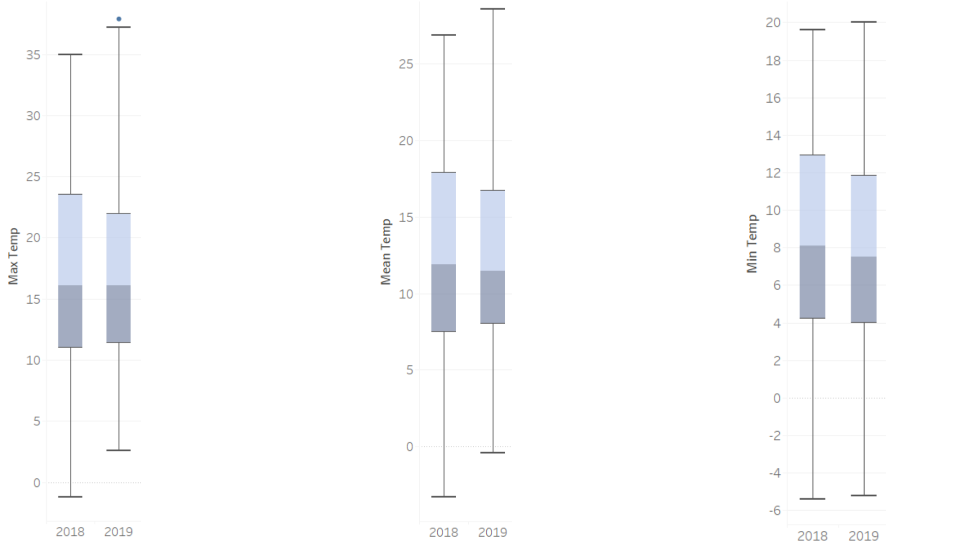
Duplicate Values

Each of the databases is analyzed to identify any duplicate values and none of the datasets contained duplicate rows.

Outliers

Outliers are data points that deviate significantly from the majority of the data, and their detection can reveal important insights or errors in the dataset. There are several ways to detect outliers. In this dissertation, to identify the outliers in the fuel price and weather dataset, box plots are used, as they are a robust tool for identifying outliers. In a box plot, outliers are detected visually as individual data points that lie beyond the whiskers. In the rentals dataset, since it had millions of records (more than 15 million), only the statistical data was analyzed.

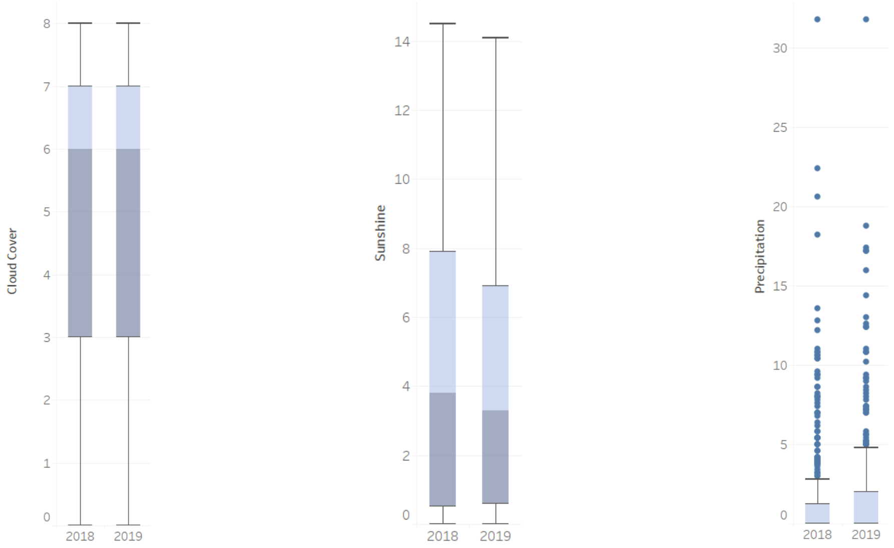
The box plot for variables *Maximum, Mean* and *Minimum Temperature*, in Figure 22, shows that the distribution is approximately symmetrical. Outliers were identified in the "Maximum Temperature" variable. These outliers were investigated to understand which specific dates they corresponded to and it was found that one of the outliers indeed represent an accurate value for a particular day, while the other was an error in the dataset and was subsequently corrected.



(a) Maximum Temperature **(b)** Mean Temperature **(c)** Minimum Temperature

Figure 22. Temperature Outliers Identification: Box plots for *Maximum, Mean* and *Minimum Temperature*

The variables *Cloud Cover* and *Sunshine* have no outliers. The same does not apply to the *Precipitation* variable (Figure 23).



(a) Cloud Cover

(b) Sunshine

(c) Precipitation

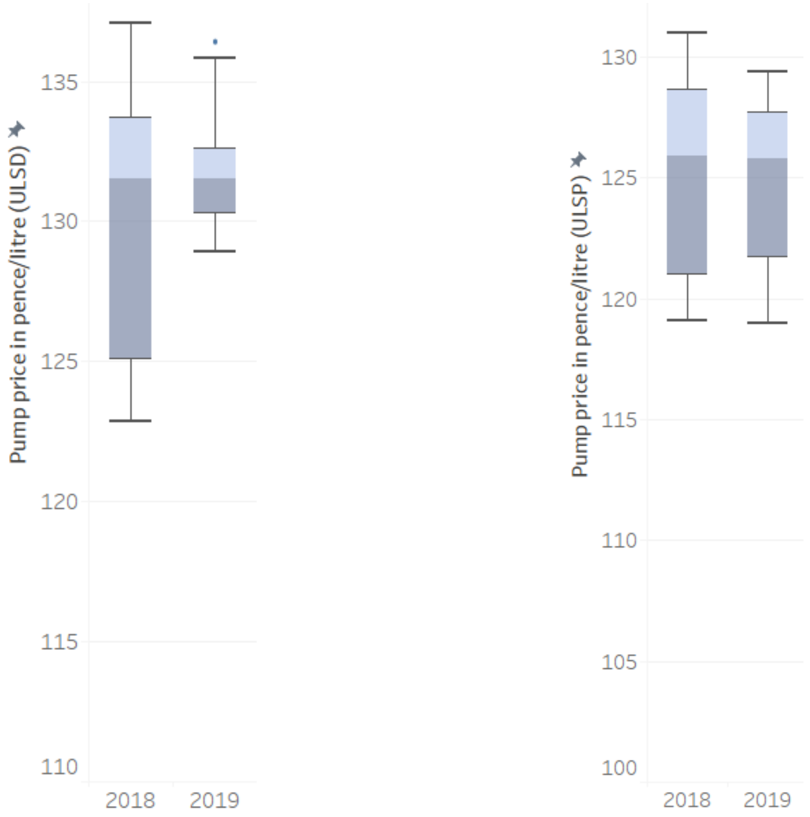
Figure 23. Weather Outliers Identification: Box plots for weather conditions

In fact, all the two years analyzed have outliers. The upper whisker limit for precipitation is less than 5mm, being 2018 a drier year with a lower upper limit, but precipitation values rising to 30mm. By observing the figure, it is possible to see that the distribution is highly skewed towards values close to zero.

The appearance of approximately 150 outliers, is indicative of extreme precipitation events. These outliers, while statistically anomalous, represents real occurrences, showing that in London, as in many other cities, it occasionally rains more than usual.

Therefore, it was decided to leave these outliers in the dataset, since they are representative of reality and, given the ongoing climate change, characterized by increasingly unpredictable weather events, their occurrence is expected to become more frequent. Hence, these outliers are significant for the intended analyses.

Figure 24 shows, once again, that although there are outliers in the ULSD Fuel Price, these represent real values and will not be changed.



(a) ULSD Fuel

(b) ULSP Fuel

Figure 24. Fuel Price Outliers Identification: Box Plots for ULSD Fuel Price and ULSP Fuel Price

As mentioned previously, because the rent variables contain millions of records, the statistical data was analyzed rather than creating box plots to examine outliers (first 4 rows of the table 4).

The analysis of the variable *Duration* reveals a very skewed distribution towards lower values. This skewness is evident as both the mean and median are considerably distant from the highest value for duration, as previously shown in Figure 15.

There are a few outliers in this variable (trips longer than 40 minutes). Since the maximum bike rental limit is 1 day, rentals longer than 24 hours were eliminated from the dataset (representing around 4400 rentals). The remaining outliers, despite having great representation, since there are just over 1 million records, remain in the dataset because they are representative of reality. Although most people use this means of transportation for short trips, there are those who prefer it for longer trips and this type of event must be considered in the analysis.

Regarding bike stations, all rentals per station were summed. For this study, only the number of bikes that left the station due to the start of people's rentals was considered. The number of bikes that arrived

Table 4: Variable's Statistical Data

Variable	**Minimum**	**Lower Whisker**	**Mean**	**Median**	**Upper Whisker**	**Maximum**	**Standard Deviation**	**First Quartile**	**Third Quartile**
Duration	1Min	1Min	19Min 50Seg	14Min	40Min 30Seg	6 Days 13H 33Min	1h 21Seg	8Min	21Min
Start Station's Rentals	4	4	24698	21233	53502	168035	17632	14509	30066
End Station's Rentals	4	4	24698	20433	54208	166217	18657	13659	29878
Daily Rental	5784	5784	28971	29348	45410	45410	8970	23791	36533
Max Temp	-1.2°C	-1.2°C	16.9°C	16°C	37.9°C	37.9°C	7°C	11.28°C	22.6°C
Mean Temp	-3.3°C	-3.3°C	12.45°C	11.8°C	28.6°C	28.6°C	5.8°C	8°C	17.1°C
Mix Temp	-5.4°C	-5.4°C	8.2°C	8.2°C	21.7°C	21.7°C	5.2°C	4.6°C	12.1°C
Precipitation	0mm	0mm	1.7mm	0mm	4.6mm	31.8mm	3.6mm	0mm	1.9mm
Sunshine	0h	0h	4.6h	3.5h	16h	16h	4.3h	0.8h	7.6h
Cloud Cover	0 oktas	0 oktas	5 oktas	6 oktas	8 oktas	8 oktas	2.4 oktas	3 oktas	7 oktas
Pump Price (ULSD)	111.7£/L	111.7£/L	127.6£/L	130.3£/L	137.08£/L	137.1£/L	6.9£/L	123.09£/L	132.39£/L
Pump Price (ULSD)	104.9£/L	107.13£/L	121.9£/L	123.8£/L	130.98£/L	131£/L	6.9£/L	119.17£/L	127.4£/L

at the station due to the returns of people's rentals was not considered. While some stations have lower rentals, others have much higher demand. As the upper whisker value is much lower than the maximum amount of rentals, it shows that certain stations have an extraordinarily high level of demand, both in the initial and final stations. To ensure that the analysis was not affected by the presence of these highly used stations, clusters were created to divide the stations based on their volume of rentals.

Inconsistencies and Errors

In terms of data inconsistencies and errors, two particular issues were identified, both in the London Weather dataset:

- The *Date* variable was formatted as YearMonthDay, which was not being recognized as a valid date. It was essential to transform this variable into the desired date format: Year - Month - Day;

- Some weather records exhibited incorrect data, specifically instances where the mean temperature exceeded the maximum temperature. These records were identified, and the data for maximum and mean temperatures were corrected using the reference data from TuTiempo (*Climate in London - Historical Weather* 2023).

4.1.2 Data Preparation

In this section, the steps regarding feature engineering, data encoding, normalization and data splitting are explained in detail.

Feature Engineering

In the feature engineering phase, the *Date Hour* variable was explored to derive several new attributes. From this variable, additional features were created, such as *Hour, Day, Year, Month, Week, Day of the Week*, and a Boolean variable indicating whether it is a weekend or not.

Data Encoding

The datasets considered for this work include both numerical and categorical variables (Table 1). To effectively use categorical variables in the intended algorithms, it was imperative to transform these variables into numerical variables – a process known as Data Encoding.

- Categorical Nominal: *Holiday, Start Station Name* and *End Station Name*. These variables have no specific order. The initial and final station variables are represented by their respective IDs, being those IDs used in the algorithms. To numerically represent the 'Holiday' variable in the dataset, an extra Boolean variable was created: it is set to 1 if it is a holiday and 0 otherwise.

- Categorical Binary Symmetric: *isHoliday* and *isWeekend*. These are binary variables and are already encoded once they are numeric values.
- Categorical Ordinal: *Month, Week, Day, Day of the Week*. These variables have a natural order, so ordinal encoding was used, where categories were replaced by integer values in an ascending order: Month - 1 to 12; Week - 1 to 52; Day - 1 to 31; Day of Week - 1 to 7;
- Numeric Variables: all the other variables. No encoding was necessary for numeric variables.

Normalization

Data normalization is an important prepossessing step for many machine learning algorithms, particularly those that are sensitive to the scale of the input features. Normalization helps to ensure that each feature contributes equally to the result and improves the convergence behavior during training. When implementing the SARIMAX and *Gradient Boosting* models, the data was not normalized. The main reason is the inherent characteristics of these models. SARIMAX relies on the actual values, trends, and seasonality of the data rather than its scale. On the other hand, *Gradient Boosting*, being a tree-based algorithm, is inherently scale-invariant. Tree-based algorithms partition the data based on feature splits that are not affected by the scale of the data. Furthermore, normalizing the data could add an unnecessary layer of complexity and processing without guaranteeing improved performance.

4.1.3 Data Splitting

When partitioning the dataset into training and testing data, which spanned two years (2018 and 2019), the initial 23 months were used as the training set to capture temporal patterns. The 24th month was designated as the testing set, allowing for an assessment of demand forecasting based on prior data. This approach ensures that predictions are evaluated against the most recent data, reflecting real-world forecasting scenarios.

4.2 Models Implementation

Since the number of bike rentals fluctuates considerably from station to station and day to day, clusters were formed to categorize the stations based on their daily rental volume. Creating clusters means grouping data into a cluster based on similarities in the data, where each cluster has data that is as similar as possible to each other but different from other clusters (Syakur et al. 2018). The clusters were created based on the median criteria, since it is less impacted by outliers.

To achieve this, the K-means clustering method was employed, once it is considered one of the most powerful clustering algorithms (Ahmed, Seraj, and Islam 2020). However, this algorithm requires the number of clusters to be defined in advance.

In order to find the best number of clusters in a K-means clustering, the Elbow method was used. This method involves the plotting of the within-cluster sum of squares (WCSS) measure for different cluster numbers and then identify the "elbow" point where WCSS starts to level off. Thus, Figure 25 shows the elbow plot, a graphic that indicates the best number of clusters for the k-means method (Syakur et al. 2018). In this analysis, cluster 6 was considered the adequate elbow point, in order to segment the data in more detail, since there are several rental stations.

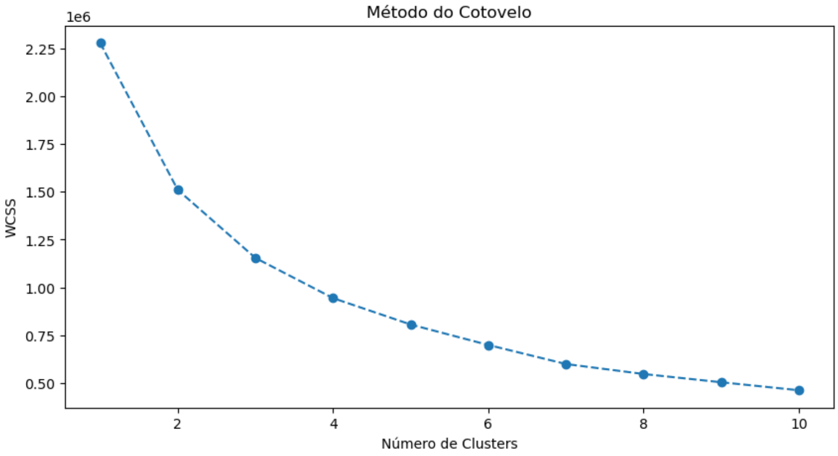


Figure 25. Elbow Method to represent the appropriate number of clusters

The statistical data of each cluster is presented in Table 5.

- Cluster 0: Presents a relatively low level of rentals. As the median is close to the mean, indicates that the data distribution is approximately symmetric. Although the maximum number of daily rentals is 174, the third quartile represents only 21 rentals, therefore most stations of this cluster have relatively low volumes of rentals. This cluster has 205 stations and some potential oultiers.

- Cluster 1: Exhibits a higher amount of rentals compared to Cluster 0. The mean and median are substantially higher at 131 and 125, respectively. It is the second cluster with most rentals, with 17 stations.

Table 5: Statistics for all clusters

Metric	**Cluster 0**	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**
Min	1	1	1	1	2	1
Mean	15	131	30	79	233	47
Median	14	125	29	76	297	47
Max	174	894	402	628	451	332
Std	9	86	15	42	128	21
1^{st} Quartile	8	76	20	53	83	33
3^{rd} Quartile	21	166	38	99	325	61
Number of Stations	205	17	311	65	2	186

- Cluster 2: Shows a moderate level of rentals, with a mean and median of 30 and 29, respectively. Once again, the third quartile is much lower than the maximum, indicating some outliars in this distribution. This cluster, composed with 311 stations, is the second with the fewest rentals, right after cluster 0.

- Cluster 3: With 65 stations, it presents a higher level of rentals, with a mean and median of 79 and 76, respectively, being the third cluster with the most rentals.

- Cluster 4: With significantly higher mean and median of 233 and 297, it presents the highest level of rentals among all clusters. It is the cluster with the greatest negative skewness, since the difference between the median and the third quartile or even the maximum value is much smaller compared to the median and the first quartile.This cluster only has 2 stations.

- Cluster 5: Has a moderate level of rental activity, with a mean and median of 47 being the third cluster with fewer rentals. This cluster includes 186 stations.

Once the clusters were formed, it was necessary to give this information to the the dataset, so an additional column was created with the cluster information.

In terms of predictive modeling, the approach shown in Figure 26 is adopted, where the SARIMAX and *Gradient Boosting* models are implemented.

The SARIMAX model is applied to a representative station from each of the six clusters. The ARIMA and SARIMA models were also applied to understand the effect of seasonality and external variables on the forecasts.

The *Gradient Boosting* model was employed using two different strategies:

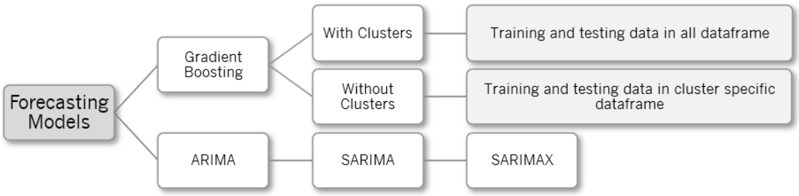

Forecasting Models
Gradient Boosting
With Clusters
Training and testing data in all dataframe
Without Clusters
Training and testing data in cluster specific dataframe
ARIMA
SARIMA
SARIMAX

Figure 26. Different Scenarios Adopted

- Universal Approach: Initially, the *Gradient Boosting* model is trained and tested using the entire dataset. This comprehensive approach aims to predict the demand at each station without any distinction or reference to its cluster.

- Cluster-specific Approach: The training and testing datasets is then partitioned based on its clusters, to understand the effect of clustering on demand forecasting.

The results of these tests will be presented on a subsequent section of this document. In this document, only the parameters for each model that yielded the best results will be presented.

4.2.1 SARIMAX Implementation

SARIMAX is a powerful algorithm capable of capturing both seasonal and non-seasonal patterns, as well as incorporate external variables into the forecast. This model has been widely recognized in the time series forecasting domain for its precision.

Since there are multiple stations and it was impractical to apply the algorithm to all of them, one station from each cluster was randomly chosen and are:

- Station ID 123 (Cluster 0)
- Station ID 217 (Cluster 1)
- Station ID 580 (Cluster 2)
- Station ID 32 (Cluster 3)
- Station ID 154 (Cluster 4)
- Station ID 41 (Cluster 5)

For the implementation of the SARIMAX model, a JupyterLab instance was used, which is an interactive web-based development environment that allows for the execution of live code, visualization of data, and textual input to document the process. To apply the SARIMAX model, it is necessary to determine the orders of the autoregressive (AR), differentiation (D) and moving average (MA) components, represented by *p*, *d* and *q*, respectively. Additionally, their corresponding seasonal orders —seasonal autoregression (*P*), seasonal differencing (*D*), and seasonal moving average (*Q*) must also be calculated. The model also requires specifying the length of the seasonal cycle (*s*), in this case 7, once the data presents weekly seasonality. For that, the *statsmodels* library was used. This library provided tools for conducting the KPSS stationarity test to verify the differentiation component, plotting autocorrelation and partial autocorrelation functions do determine autoregressive and moving average components, and implementing both ARIMA and SARIMAX models for time series analysis.

Data Pre-Processing

As explained in the Litherature Revew's section, it is crucial to check the stationarity of the series, namely the variable *Rentals Quantity*, to ensure that the data is suitable for the model in question.

To verify the stationarity of the data, the Kwiatkowski-Phillips-Schmidt-Shin (KPSS) test is used. This test examines whether the series is stationary around a deterministic trend. In this test, each entry represents the number of rentals at a specific station on a given day. The KPSS test is then executed using *statsmodels* python library (used for statistical analysis and data modeling), which provides some statistical data along with a *p-value*. By comparing the *p-value* to a predetermined significance level, typically 0.05, it is possible to determine if the data exhibit stationarity. If the p-value is less than 0.05, the null hypothesis of stationarity is rejected, indicating the presence of non-stationarity in the dataset

Based on the *p-value* of the tests, that are listed bellow, the necessary differences to achieve stationarity are applied.

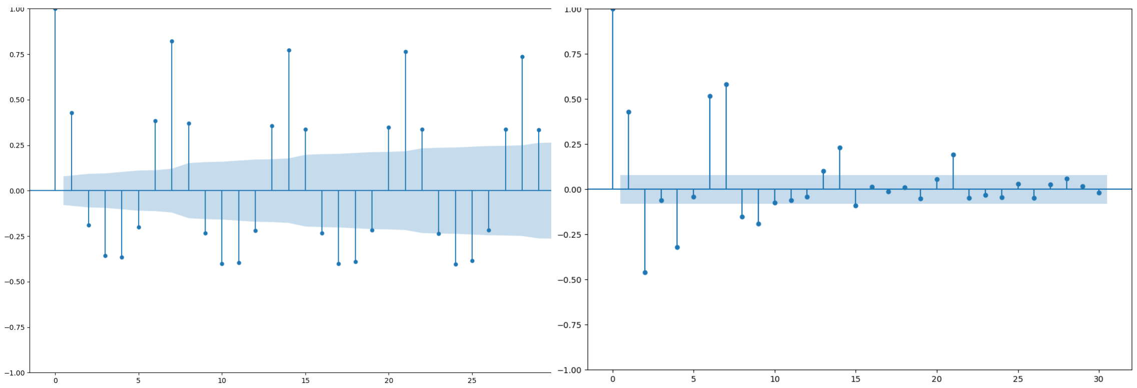
- Station ID 32: Data was non-stationary. It was necessary to differenciate the data in order to turn it into a stationary variable.
- Station ID 41: Data was already stationary. It was not necessary to differentiate the data.
- Station ID 123: Data was non-stationary. It was necessary to differenciate the data in order to turn it into a stationary variable.
- Station ID 154: Data was already stationary. It was not necessary to differentiate the data.
- Station ID 217: Data was non-stationary. It was necessary to differenciate the data in order to turn it into a stationary variable.

- Station ID 580: Data was non-stationary. It was necessary to differenciate the data in order to turn it into a stationary variable.

Parameter Selection

To determine the optimal *p* and *q* parameters for this model, the Autocorrelation function (ACF) and Partial Autocoorelation function (PACF) are used. These functions help to get an initial idea of potential *p* and *q* values. The *q* is identified by the number of the lags when ACF graph reaches the confidence interval for the first time and the *p* follows the same reasoning but for PACF graph.

For the specific example of station ID 41 (Figure 27), the *q* is identified by the number of the lags when ACF reaches the confidence interval, indicated by the blue area, for the first time, in this case, lag 2 (the first one is lag 0) and the *p* follows the same reasoning but for PACF graph, in this case, lag 2. This indicates a potential AR (autoregressive) term of 2, an MA (moving average) term of 2 and a differentiation term of 0 since none differentiation was required. Nonetheless, the best results may come from different values of q or p, with this graphs being just a way of helping to define them.



(a) ACF Graph for Station ID 41

(b) PACF Graph for Station ID 41

Figure 27. ACF and PACF for SARIMAX Parameter Selection

Model Building and Evaluation

Subsequently, a grid search is implemented to identify the combination of parameters that minimized the errors. Although, for comparison purposes, all three algorithms related to ARIMA were implemented, only the best parameters for the SARIMAX model are presented, as it is the algorithm being compared with *Gradient Boosting* and is expected to provide better predictions than ARIMA and SARIMA. Nonetheless, the results for each of the three algorithms are presented in the subsequent chapter.

The parameters are structured as follows: (p, d, q) (P, D, Q, s). Note that when the SARIMA amd

ARIMA models were implemented, the parameters used were others, as they created better forecasts.

The best combination of parameters for each station is:

- Station ID 123: (3, 1, 1) (1, 0, 0, 7)
- Station ID 217:(2, 1, 1) (1, 0, 1, 7)
- Station ID 580: (6, 1, 2) (0, 0, 0, 7)
- Station ID 32: (1, 1, 2) (1, 0, 1, 7)
- Station ID 154: (1, 0, 1) (2, 0, 1, 7)
- Station ID 41: (2, 0, 1) (2, 0, 1, 7)

4.2.2 Gradient Boost Implementation

In this research, the *Gradient Boosting* algorithm is applied using xgBoost framework, a well-known framework for this type of forecasting. XGBoost has gained wide recognition in the machine learning community for its performance and scalability, without the need for very powerful computing resources.

Hyperparameter Tuning

For the fine-tuning of the XGBoost model, several hyper parameters are considered (*XGBoost Documentation* 2023). In each iteration, different values for these parameters are explored to optimize the model's performance.

- Learning Rate: Defines the step size at each iteration while moving towards a minimum of the loss function.
- Max Depth: Determines the depth of the tree. It is used to control over-fitting as higher depth will allow the model to learn very specific relations for a particular sample.
- Subsample: Denotes the fraction of observations that are randomly sampled for each tree.
- Colsample by tree and Colsample by level: These parameters control the subsampling of columns at the global level and at the level-wise growth respectively.
- Gamma: Specifies the regularization in the leaves and the minimum loss reduction required for a split to occur.
- Min Child Weight: Defines the minimum sum of instance weight needed in a child.

Parameter	**Cluster 0**	**Cluster 1**	**Cluster 2**	**Cluster 3**	**Cluster 4**	**Cluster 5**
learning_rate	0.01	0.1	0.1	0.08	0.15	0.01
max_depth	10	10	7	9	9	9
subsample	0.9	0.8	0.9	0.8	0.7	0.9
colsample_bytree	0.85	0.9	0.8	0.9	0.7	0.8
colsample_bylevel	0.85	0.7	0.7	0.7	0.6	0.7
gamma	0	0	0	0	0	0
min_child_weight	0.8	1	1	1	0.7	1
reg_alpha	0.05	0.5	0.04	0.04	0.04	0.05
reg_lambda	1	0.1	1	1	1	1
n_estimators	30000	5000	5000	8000	8000	20000
objective	reg:squarederror	reg:squarederror	reg:squarederror	reg:squarederror	reg:squarederror	reg:squarederror

Table 6: Model parameters for each Clusters

- Regularization *Alpha* and Regularization *Lambda*: These are L1 and L2 regularization terms in the weights, respectively.

- Number of estimators: Denotes the number of gradient boosted trees to be constructed.

- Objective: Determines the loss function to be minimized.

Once again, during tuning of the model, an extensive grid search was applied to find the best combination of hyperparameters. For each cluster, the best combination is given by Table 6.

Chapter 5

Results

This chapter presents the evaluation metrics for each of the implemented forecasting models, in order to determine which is the best fit for this problem.

5.1 Algorithms Performance

In the assessment of the prediction models, various metrics were meticulously examined to evaluate their performance, namely RMSE, MAE, MAPE, MSE, RMSLE and $\mathbb{R}^2$. The evaluation metrics were calculated using the *sklearn* library, where the best metric is identified by the lowest values, except for the $\mathbb{R}^2$, where a higher value indicates better results.

Table 7 compares the results of the two strategies used to test the *Gradient Boosting* algorithm: the universal approach and the cluster-specific approach, where the first column identifies the algorithm ID, the second one refers to the strategy used, the third indicates the cluster used in the training/testing sets and the last six columns show the values obtained for each evaluation metric. The evaluation metrics refers to the model that was trained and tested based on stations grouped by each cluster, rather than testing the model on individual stations.

Table 7: *Gradient Boosting* algorithm evaluation metrics

Alg. ID	**Approach**	**Cluster**	**RMSE**	**MAE**	**MAPE**	**MSE**	**RMSLE**	$\mathbb{R}^2$
1	Universal Approach	All Clusters	8.97	6.47	33.72	80.74	0.35	0.93
2	Cluster-specific Approach	Cluster 0	4.27	3.31	40.93	18.23	0.37	0.87
3	Cluster-specific Approach	Cluster 1	19.76	14.59	20.50	390.44	0.26	0.98
4	Cluster-specific Approach	Cluster 2	6.84	5.30	27.06	46.73	0.31	0.80
5	Cluster-specific Approach	Cluster 3	12.43	9.49	23.87	154.46	0.27	0.97
6	Cluster-specific Approach	Cluster 4	24.71	17.93	13.95	610.35	0.17	0.99
7	Cluster-specific Approach	Cluster 5	8.61	6.65	21.20	74.20	0.25	0.88

Checking the metrics of the algorithm applied in the universal approach, the results seem to be quite good. In order to compare which approach would be the most suitable, both approaches were applied to each of the stations and the results were as follows:

- Cluster 0: Cluster-specific approach presented better results in about 75% of the stations;
- Cluster 1: Cluster-specific approach presented better results in about 53% of the stations;
- Cluster 2: Cluster-specific approach presented better results in about 63% of the stations;
- Cluster 3: Cluster-specific approach presented better results in about 75% of the stations;
- Cluster 4: Cluster-specific approach presented better results in about 50% of the stations;
- Cluster 5: Cluster-specific approach presented better results in about 80% of the stations;

For all clusters, the model performs better when the training and testing data are confined to its Cluster, instead of the entire dataset, so the approach most recommended for obtaining the best forecasts would be the cluster-specific approach.

Regarding ARIMA modeling and its derivations (Table 8), it is evident that the SARIMAX version outperforms the other models (ARIMA or SARIMA), with its intrinsic ability to account for seasonality, especially given the clear seasonal trends present in the dataset. Moreover, SARIMAX's ability to integrate external variables offers a more holistic model, capturing influences that the standard ARIMA might ignore.

5.2 Forecast Results

Table 9 presents the metrics of the forecast evaluation for the algorithms SARIMAX and *Gradient Boosting* when applied to each of the stations selected for each cluster. Only these two algorithms are represented in the table because they were the ones that presented the best results.

In general, although all algorithms perform well, *Gradient Boosting* algorithms show slightly better results in all metrics when compared to SARIMAX, except for Station ID 123 and Station ID 580. For Station ID 123, each one of the algorithms shows better results in three different metrics. Analyzing each metric in detail, *Gradient Boosting* seems to have larger errors - higher MSE and RMSE where larger errors have greater impact on the result. For this specific station, both algorithms could be chosen as there is no one that stands out from the other, it would depend on the metrics that the owner of the system would

Table 8: ARIMA, SARIMA and SARIMAX algorithms evaluation metrics

Alg. ID	**Method**	**Station ID**	**RMSE**	**MAE**	**MAPE**	**MSE**	**RMSLE**	$\mathbb{R}^2$
8	ARIMA	Station ID 32	14.60	11.58	25.74	213.74	0.27	0.75
9	SARIMA	Station ID 32	12.00	9.85	20.56	144.07	0.22	0.80
10	SARIMAX	Station ID 32	11.51	9.08	18.23	132.53	0.17	0.83
11	ARIMA	Station ID 41	7.8	5.78	18.35	60.84	0.21	0.93
12	SARIMA	Station ID 41	6.23	4.98	20.90	38.90	0.28	0.96
13	SARIMAX	Station ID 41	6.04	5.08	17.04	36.43	0.23	0.96
14	ARIMA	Station ID 123	4.86	3.63	25.43	23.66	0.28	0.62
15	SARIMA	Station ID 123	4.54	3.36	22.36	19.21	0.25	0.67
16	SARIMAX	Station ID 123	4.13	3.20	19.72	17.03	0.22	0.71
17	ARIMA	Station ID 154	33.67	28.99	38.90	1133.90	0.41	0.93
18	SARIMA	Station ID 154	26.37	21.45	25.66	701.62	0.98	0.90
19	SARIMAX	Station ID 154	25.50	19.42	24.44	650.47	1.04	0.96
20	ARIMA	Station ID 217	22.34	16.61	20.96	499.22	0.24	0.86
21	SARIMA	Station ID 217	20.22	14.81	20.99	408.80	0.24	0.90
22	SARIMAX	Station ID 217	17.15	12.64	17.07	294.00	0.22	0.93
23	ARIMA	Station ID 580	4.68	3.87	15.91	21.88	0.18	0.25
24	SARIMA	Station ID 580	4.43	3.61	13.97	20.72	0.18	0.53
25	SARIMAX	Station ID 580	4.38	3.45	12.77	19.17	0.18	0.61

give more importance to. Regarding Station ID 580, the MAPE metric presents a slightly higher value in the Gradient Boost algorithm, as well as the R2 metric, which has a worse result.

Nonetheless, in all the tests carried out, all the algorithms presented very similar results, which leads to the conclusion that regardless of the choice of the best algorithm, any one that was implemented would result in good predictions.

Figures 28 29, 30, 31, 32 and 33 illustrates a visual demonstration of the forecasts implemented.

Table 9: Forecast evaluation metrics

Alg. ID	**Station ID**	**Algorithm**	**RMSE**	**MAE**	**MAPE**	**MSE**	**RMSLE**	$\mathbb{R}^2$
5	Station ID 32	*Gradient Boosting*	7.09	6.28	12.27	62.50	0.14	0.91
10	Station ID 32	SARIMAX	11.51	9.08	18.23	132.53	0.17	0.83
7	Station ID 41	*Gradient Boosting*	5.82	4.53	10.66	33.91	0.13	0.96
13	Station ID 41	SARIMAX	6.23	4.98	20.90	38.90	0.28	0.96
2	Station ID 123	*Gradient Boosting*	4.30	2.84	14.57	18.47	0.19	0.68
16	Station ID 123	SARIMAX	4.13	3.20	19.72	17.03	0.22	0.71
6	Station ID 154	*Gradient Boosting*	21.93	16.62	13.45	481.25	0.16	0.97
19	Station ID 154	SARIMAX	25.50	19.42	24.44	650.47	1.04	0.96
3	Station ID 217	*Gradient Boosting*	14.00	12.35	14.77	195.84	0.17	0.95
22	Station ID 217	SARIMAX	17.15	12.64	17.07	294.00	0.22	0.93
4	Station ID 580	*Gradient Boosting*	3.86	3.25	12.92	14.94	0.15	0.48
25	Station ID 580	SARIMAX	4.38	3.45	12.77	19.17	0.18	0.61

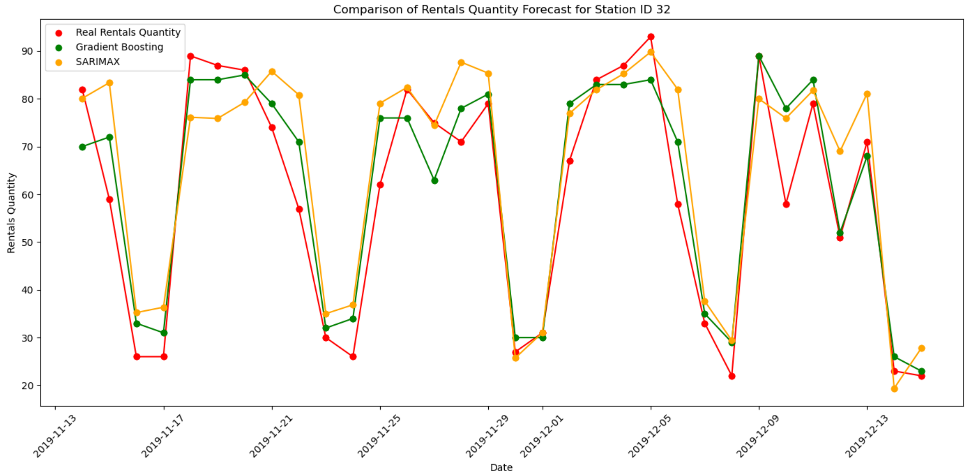


Figure 28. Forecast of all algorithms for Station ID 32

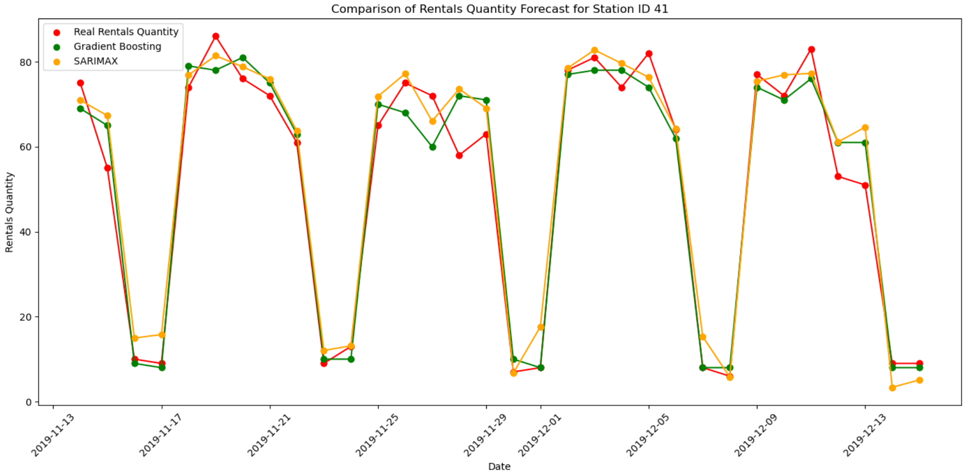


Figure 29. Forecast of all algorithms for Station ID 41

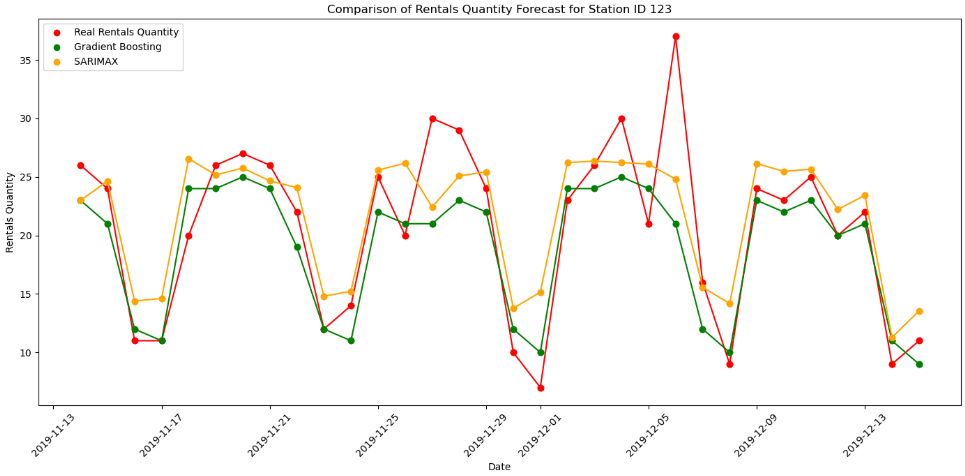


Figure 30. Forecast of all algorithms for Station ID 123

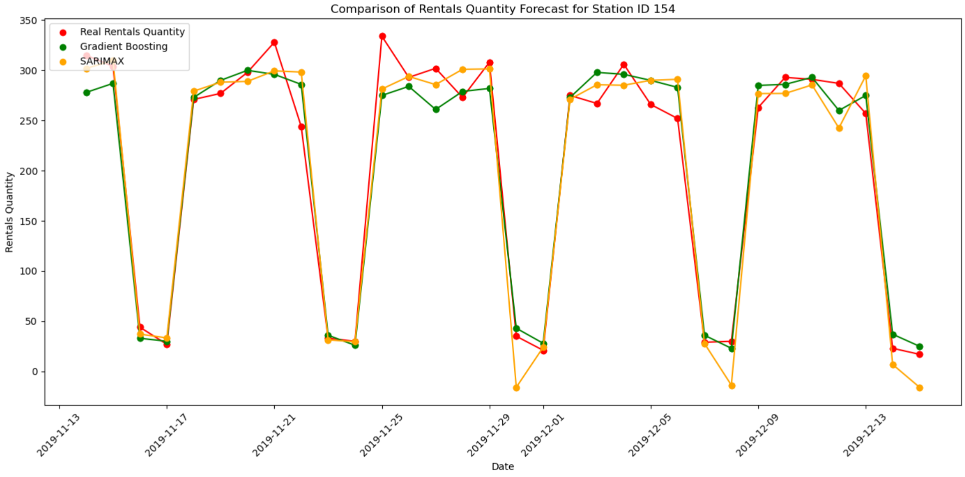


Figure 31. Forecast of all algorithms for Station ID 154

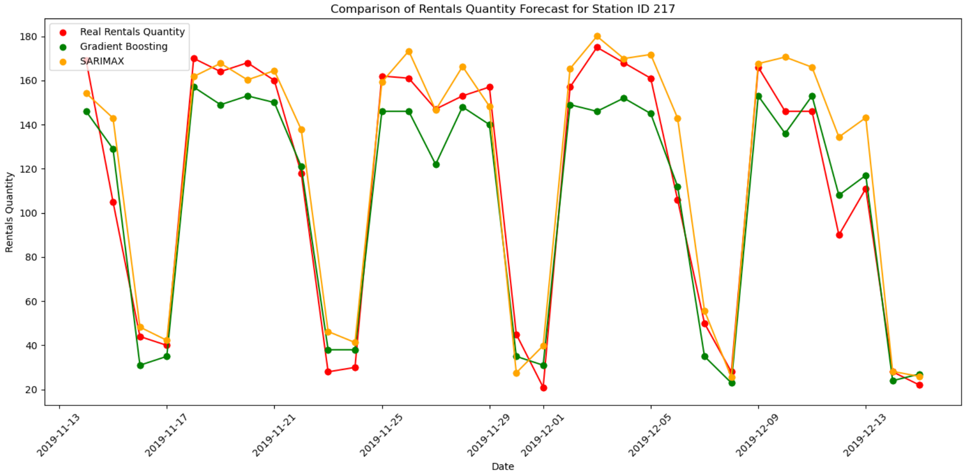


Figure 32. Forecast of all algorithms for Station ID 217

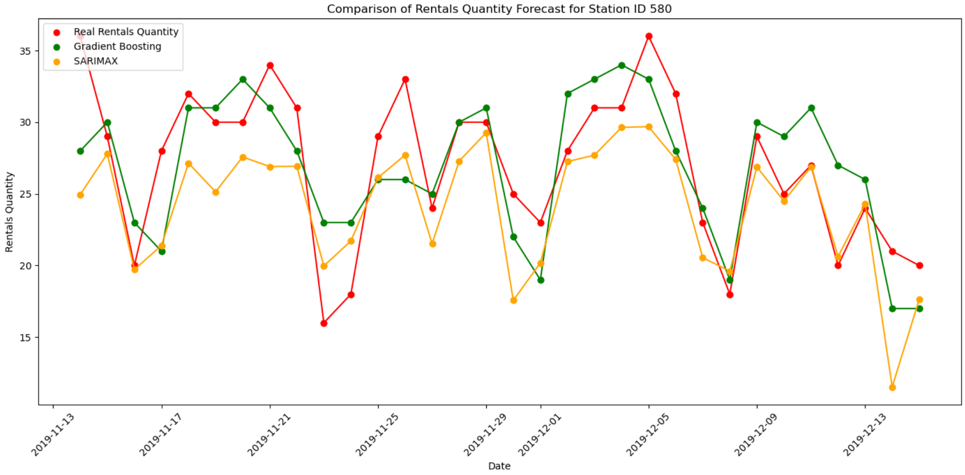


Figure 33. Forecast of all algorithms for Station ID 580

Chapter 6

Conclusions and Future work

6.1 Conclusions

Population growth has brought several challenges in the field of urban mobility, forcing governments and companies to rethink their transport infrastructure strategy, investing in more sustainable and efficient solutions. One of these solutions is the use of alternative means of transport, such as bikes. In this sense, more and more cities have implemented bike-sharing systems.

That way, bike rental has become an increasingly popular mode of transportation, bringing with it some challenges, such as the correct allocation of bikes among the different stations or accurately forecast the bike rentals. Also, adequate infrastructure, integration with other modes of transportation, theft security, and user-friendly software are essential elements for their success.

This dissertation is focused on the development of bike rental forecasting algorithms, more specifically, the daily rental forecasts at each rental station. To achieve this, a database containing rental information was used, which was complemented with data that could influence these rentals, including weather data, fuel costs, and holidays and festive days. Therefore, the final database used in this study contained data such as the start and end date and time of each rental, the departure and arrival stations, the duration of the trip, and the bike ID, which were complemented with fuel prices for each of the days analyzed, as well as their temperature, precipitation, hours of sunshine, cloud cover, and so on. Lastly, it included indications of holidays occurring during the years analyzed.

One of the key steps to improve the forecasts involves analyze and understand the data to identify patterns within it. The findings reveal that temporally, on weekdays, there is a rental increases during morning, from 6am to 9am, and evening from 4pm to 7pm peak hours, suggesting a strong alignment with commuting times. At the weekend, rentals showed an homogeneous increase in demand between mid-morning and late afternoon, which may indicate recreational or leisure activities. Furthermore, rentals are higher during weekdays compared to weekends, which reinforces the use of this type of transport for commuting.

Geographical patterns were also observed highlighting the importance of location. The high number of rentals around Hyde Park and City of London demonstrates the impact of recreational and tourist areas on bike rentals, as well as work hubs. Furthermore, the number of rentals in stations near transport hubs such as Waterloo and King's Cross stations suggests that bikes often complement other types of transportation.

The short duration of most bike rides, with a peak of less than 2 minutes, indicates a preference for using bikes for quick transfers within the city.

Before proceeding with the development of the forecasting models, it was necessary to pre-process the data, so that they had the desired quality, and for this the missing data were filled in, the outliers were analyzed to understand what should be done with them, since these may be errors or real data, and data inconsistencies and errors has been corrected.

From a forecasting perspective, it was tested two approaches of models, a more traditional one, with ARIMA and its derivations, such as SARIMA and SARIMAX, as well as a more advanced approach, where the *Gradient Boosting* algorithm was applied. In this last one, to understand the impact that stations with higher or lower rentals would have on the forecastings, clusters of stations were created based on their rental volumes. To evaluate the forecast results, one station was randomly selected from each cluster, as presenting the data for each station would be impractical.

When exploring the ARIMA and its derivations, the SARIMAX model showed better results, given the clear seasonal nature present in the dataset. In addition, the ability of SARIMAX to integrate external variables provides a more holistic model, capturing influences that the standard ARIMA may not be able to. Nonetheless, the results obtained from both the ARIMA and SARIMA models were reasonably satisfactory, with the latter outperforming the former. Despite this, SARIMAX did not produce predictions as reliable as the *Gradient Boosting* model.

Among the various approaches tested regarding *Gradient Boosting* techniques, the results showed that predictions seems to benefit from training and testing the data on cluster-specific data and not using the entire dataset. This can be explained by the fact that there is a large set of stations, some of which being quite heterogeneous among themselves, and by grouping stations into clusters that have similar observations of rentals, it can help the model to capture more precise and specific patterns present in each cluster.

In summary, although all the algorithms showed good predictions, the approach that stands out positively is the *Gradient Boosting* algorithm, more precisely when stations are divided into clusters based on rental quantities. However, within the implemented set of algorithms, the system would benefit from accurate predictions regardless of the implemented algorithm.

6.2 Prospect for Future work

Based on the highlights of this research, there are some key points that could be analyzed as future work in the domain of bike sharing systems:

- Impact of COVID-19 on forecasts: The global pandemic has affected various sectors, including urban mobility. It would be interesting to study the impact of COVID-19 on bike rentals and analyze whether the model created reflects the changes brought by the pandemic. If necessary, the models created could be refined to take into account that changes in patterns.
- Route Predictions: Beyond rental forecasts for each station, predicting potential bike routes could provide valuable insights, since it can help to rethink the strategy for distributing bikes across stations, ensuring optimal availability.

Bibliography

Aamer, Ammar Mohamed, Luh Putu Eka Yani, and I Made Alan Priyatna (2021). “Data Analytics in the Supply Chain Management: Review of Machine Learning Applications in Demand Forecasting”. In: *OPERATIONS AND SUPPLY CHAIN MANAGEMENT* 14.1, pp. 1–13.

Ahmed, Mohiuddin, Raihan Seraj, and Syed Mohammed Shamsul Islam (2020). “The k-means Algorithm: A Comprehensive Survey and Performance Evaluation”. In: *Electronics* 9.8.

Armstrong, J. Scott (1988). “Research Needs In Forecasting”. In: *International Journal of Forecasting* 4, pp. 449–465.

Babaee Tirkolaee, Erfan et al. (June 2021). “Application of Machine Learning in Supply Chain Management: A Comprehensive Overview of the Main Areas”. In: *Mathematical Problems in Engineering* 2021, pp. 1–14.

Badulescu, Yvonne, Ari-Pekka Hameri, and Naoufel Cheikhrouhou (Feb. 2021). “Evaluating demand forecasting models using multi-criteria decision-making approach”. In: *Journal of Advances in Management Research* ahead-of-print. DOI: 10.1108/JAMR-05-2020-0080.

Barrera-Animas, Ari Yair et al. (2022). “Rainfall prediction: A comparative analysis of modern machine learning algorithms for time-series forecasting”. In: *Machine Learning with Applications* 7.

Bills, John (Apr. 2023). *19 cities with the best public transport in the world - according to locals*. TimeOut. URL: https://www.timeout.com/travel/best-public-transport-in-the-world.

Brownlee, Jason (Aug. 2016). *XGBoost With Python*. 1st ed. Gradient Boosted Trees with XGBoost and scikit-learn. Machine Learning Mastery.

Chopra, Sunil and Peter Meindl (2012). *Supply Chain Management: Strategy, Planning, and Operation*. 5th ed. Kellogg School of Management and Kepos Capital: Pearson.

Climate in London - Historical Weather (2023). URL: https://en.tutiempo.net/climate/02-2020/ws-37720.html.

DeMaio, Paul J. (2003). *Smart Bikes: Public Transportation for the 21st Century*. Vol. 57. 1, pp. 9–11.

Deng, Haowen et al. (Dec. 2021). “Ensemble learning for the early prediction of neonatal jaundice with genetic features”. In: *BMC Medical Informatics and Decision Making* 21.

ECA&D (2023). *European Climate Assessment Dataset*. URL: https://www.ecad.eu/.

Elshewey, Ahmed M. et al. (2023). “A Novel WD-SARIMAX Model for Temperature Forecasting Using Daily Delhi Climate Dataset”. In: *Sustainability* 15.1.

European Commission (2021). *Facts and Figures Cyclists*. Tech. rep. Brussels, European Commission, Directorate General for Transport: European Road Safety Observatory.

Hernández, Diego (Aug. 2017). “Public transport, well-being and inequality: coverage and affordability in the city of Montevideo”. In: *CEPAL Review* 122, pp. 151–169.

Hyndman, Rob J. and George Athanasopoulos (May 2021). *Forecasting: Principles and Practice*. 3rd ed. A comprehensive introduction to the latest forecasting methods using R. Learn to improve your forecast accuracy using dozens of real data examples. Texts.

IBM (2023). *CRISP-DM Help Overview*. URL: https://www.ibm.com/docs/de/spss-modeler/18.1.1?topic=dm-crisp-help-overview.

Kocaoglu, Batuhan, A. Zafer Acar, and Behlül Yılmaz (June 2014). “Demand Forecast, Up-to-date Models, and Suggestions for Improvement: An Example of a Business”. In: *Journal of Global Strategic Management* 8.1, pp. 26–37.

Kuhn, Max and Kjell Johnson (Mar. 2018). *Applied Predictive Modeling*. 2nd ed. Springer.

Saltz, Jeffrey S., Ivan Shamshurin, and Kevin Crowston (2017). “Comparing Data Science Project Management Methodologies via a Controlled Experiment”. In: *Proceedings of the 50th Hawaii International Conference on System Sciences*.

Saupin, RGuillaume (Oct. 2022). *Practical Gradient Boosting: A deep dive into Gradient Boosting in Python*. 1st ed. AFNIL.

Schröer, Christoph, Felix Kruse, and Jorge Marx Gómez (2021). “A Systematic Literature Review on Applying CRISP-DM Process Model”. In: *Procedia Computer Science* 181, pp. 526–534.

Syakur, M A et al. (Apr. 2018). “Integration K-Means Clustering Method and Elbow Method For Identification of The Best Customer Profile Cluster”. In: *IOP Conference Series: Materials Science and Engineering* 336.1.

Torres, José et al. (Dec. 2020). “Deep Learning for Time Series Forecasting: A Survey”. In: *Big Data* 9.

Transport for London (2023). *Santander Cycles*. URL: https://tfl.gov.uk/modes/cycling/santander-cycles.

UK Government (2023). URL: https://www.gov.uk/government/statistical-data-sets.

UnitedNations (2023). *Urbanization*. URL: https://www.un.org/development/desa/pd/content/urbanization-0 (visited on 2023).

XGBoost Documentation (2023). URL: https://xgboost.readthedocs.io/en/stable/.