LEVERAGING CLUSTERING FOR BOOKING PATTERN FORECASTING: A CASE STUDY

Anni Tziakouri (13987151)
BSc Business Analytics, Universiteit van Amsterdam

UNIVERSITY OF AMSTERDAM

A Bachelor thesis conducted with Sunweb Group
Supervisor : Dr. Kevin Pak
27 of June 2024

ABSTRACT

Accurately forecasting the timing of future bookings is crucial for effective revenue management and strategic planning for holiday package providers. This thesis focuses on forecasting booking patterns utilizing data from Sunweb, a leading holiday package provider in Europe. Due to the considerable fluctuation in booking behavior, forecasting models may struggle to capture all the underlying trends. To address this, this thesis employs clustering to identify and group similar booking patterns, enabling a deeper understanding and analysis. It further evaluates the impact of incorporating clustering into booking pattern forecasting through ensemble modeling. The premise is that training a forecasting model on groups of similar patterns is expected to enhance the model's ability to accurately capture these trends. Six different ensembles were built, using three different clustering models and were compared with four general forecasting models and a benchmark model. Five out of the six ensemble models either outperformed or matched the benchmark model's performance, while the general forecasting models, trained on all the booking patterns, did not surpass the benchmark model. These findings indicate that the ensemble models achieved the lowest forecasting errors, underscoring the significant role clustering plays in enhancing forecasting accuracy.

.

TABLE OF CONTENTS

CHAPTER

1. Introduction

Over the years, there has been a growing interest in forecasting models, as forecasting plays a crucial role in strategy and budget planning within organizations (Makridakis, 1996; Taylor and Letham, 2018). For companies, accurate forecasts can lead to better revenue management, since these forecasts are a major input to most revenue management systems (L. Weatherford & Kimes, 2003).

This research utilized data collected from Sunweb, a leading holiday package provider in Europe (Sunweb Group, 2023). Sunweb pre-purchases airline tickets and reserves hotel rooms, namely their holiday package capacity, in advance. If they purchase a capacity of 100 packages, they must sell exactly 100 packages; selling fewer would result in financial losses. Given this situation, Sunweb already knows in advance their demand targets and therefore demand forecasting is redundant. Hence, Sunweb focuses on creating forecasts to monitor weekly bookings per holiday package i.e. booking patterns, that can provide at any given point in time what percentage of package capacity should be sold. By monitoring these forecasts, Sunweb can respond to variations in their performance. For instance, if for a given holiday package 90% of the capacity is projected to be reached 2 weeks before departure but is only at 70%, then Sunweb can respond by lowering the prices to mitigate losses or boosting their marketing spend.

The challenge lies in accurately forecasting the booking patterns of each holiday package, since there is considerable fluctuation in booking behavior (Webb et al., 2020). This behavior can be influenced by several factors, such as preferences regarding booking timing and seasonal trends. Thus, booking patterns can vary widely: some holiday packages are reserved far in advance, others show consistent weekly bookings, and some experience a surge of last-minute reservations. Seasonal booking trends also play a significant role, with peak booking periods typically occurring post-holidays, alongside medium and slow seasons. All of these factors, add another layer of complexity to forecasting.

Existing research on forecasting has mainly focused on the airline and hotel industry. Different methods include time series analysis, econometric models, booking pattern clustering, Machine Learning (ML) or combinations of the above (De Sá, 1987; Makridakis et al., 1982; Masini et al., 2021; Schwartz and Hiemstra, 1997; Viverit et al., 2023). No consensus has been reached on the optimal method for achieving superior results. There are a lot of contradicting studies in the field; for instance Webb et al. (2020) argued the benefit of using Neural Networks in forecasting while Faraway and Chatfield (1998) concluded that it was ineffective. The general conclusion is that the most accurate methods will vary by context (L. Weatherford & Kimes, 2003), forecasting horizon (Makridakis et al., 2020) and even the error measure used (Makridakis et al., 1982).

Moreover, according to Schwartz (2008), the varying booking behaviors, as shown by the distribution of past booking patterns, significantly influences the forecasting of future patterns. Building on this, subsequent research by Bandara et al. (2020) and Viverit et al. (2023) explored different clustering methods to identify similar patterns in bookings, proving that by understanding and integrating them into model training, can indeed enhance the forecasting accuracy. The idea is that training the forecasting models at the cluster level, where more similar booking patterns exist, can allow them to better capture the underlying trends. In Sunweb's context, uncovering these trends can also provide insights into the diverse booking behaviors mentioned; for example that some holiday packages may experience earlier reservations compared to others, while others may exhibit consistent bookings or even steep increases closer to the departure.

This thesis aims to make a novel contribution in the literature of booking pattern forecasting by comparing different clustering methods and examining their impact on the forecasting performance. Firstly, it synthesizes existing literature on clustering time-series data, to identify the most effective approach for grouping booking patterns. Secondly, it investigates whether clustering can reveal distinct underlying booking trends and create meaningful groupings of the holiday packages offered. Thirdly, this thesis evaluates whether the integration of clustering into forecasting models can enhance the forecasting accuracy, potentially leading to the development of a more robust model.

Hence, this thesis will evaluate the performance of various forecasting models, namely ensemble models trained on clustered data and other forecasting models, offering practical guidance on selecting the optimal forecasting model. As such, the research question is: To what extent can clustering methods improve the forecasting accuracy of holiday packages' booking patterns?

2. Literature Review

2.1 Forecasting Methods

Forecasting models are crucial for organizations for optimizing revenue, as accurate forecasts serve as pivotal inputs for most revenue management systems (Tse & Poon, 2015). Widely-used forecasting models have been developed and can be further subcategorized into three categories: historic, advanced and combined booking models (L. Weatherford & Kimes, 2003). The techniques used therein for forecasting have primarily focused on time-series, econometric and machine learning approaches (Banerjee et al., 2020).

Historic booking models, which will be employed in this thesis, rely on past booking data to forecast future bookings (L. Weatherford & Kimes, 2003). This approach assumes that variations in bookings follow cyclical and seasonal patterns and therefore the previous patterns can be used to forecast the future ones. These models will be employed in this thesis, since the booking pattern forecast for each holiday package requires to be made in advance.

For historic booking models, time series techniques such as autoregressive models, for example ARIMA, and exponential smoothing have been used. Their aim is to predict the distribution of the final bookings by modeling the underlying trends and seasonality (Nieto & Carmona-Benítez, 2018). The main difference is in their approach of incorporating past data. ARIMA (Autoregressive Integrated Moving Average), forecasts future values based on past observations by combining autoregression, differencing, and moving averages, emphasizing observations with high autocorrelation (Geurts et al., 1977). Exponential smoothing assigns weights to each observation, with higher weights being given to more recent ones, to calculate a weighted moving average, providing smoother values for each period to handle trends and seasonality (Ahmad & Ahmad, 2013).

Ahmad and Ahmad (2013) conducted a comparative analysis of ARIMA and exponential smoothing, which they identified as two of the fastest-growing models in the field of time series forecasting. This study focused on varying forecasting horizons and different volumes of data, demonstrating that the ARIMA model was more effective for long-term forecasts, even with limited data. Conversely, Exponential Smoothing excelled in predicting time series with minimal variation between consecutive points, but was less reliable for long-term predictions, potentially due to the error accumulation.

Nevertheless, Makridakis et al. (1982) in their M competition, suggested that for historic models, more complex and statistically sophisticated methods, such as ARIMA, generally do not outperform simpler ones. This aligns with De Sá (1987), who initially employed an ARIMA model for fare-

specific forecasts; however, finding it ineffective, shifted to an econometric model of linear regression, aiming to understand the causality between explanatory variables and the forecasted bookings. Econometric models such as linear regression, though, are sometimes limited in their ability to capture complex and non-linear relationships, and consequently new ML methods have emerged to address the irregularities and volatility in historical bookings (Ahmed et al., 2010). For further discussion refer to *section 2.2*.

The second category of forecasting models is advanced booking modeling, which involves considering the bookings already made, commonly referred to as bookings-on-hand, for a specific departure and then extrapolating the total bookings (L. Weatherford & Kimes, 2003). The main difference with historic models is that advanced models consider the build-up of the current reservations and ignore the past booking data. Advance booking modeling is a well-documented area since it is particularly applicable for forecasting when there is little, or none, historic data available. It is typically applied to short-term, close-to-departure forecasting (Tse & Poon, 2015).

L. Weatherford and Kimes (2003) conducted a comparison of some commonly-used advanced forecasting models: (logarithmic) linear regression, additive and multiplicative models and two smoothing techniques, between different length of stays and rate categories to Choice Hotels. In additive models (also referred to as "pick-up" models) the number of bookings-on-hand is considered independent of the final bookings, while multiplicative models assume that the future reservations are influenced by the current number of bookings-on-hand. L. Weatherford and Kimes (2003) demonstrated how to quantify relationships between current bookings and final bookings and their findings revealed that the additive model and regression exhibited the most robust performance for advanced models. Similar to this research, Wickham (1995) presented a simple linear regression using the bookings-on-hand, which was in this case outperformed by an advanced additive model.

However, Tse and Poon (2015) discovered trends in past booking patterns, challenging the industry's reliance only on bookings-on-hand, without considering the past booking data and how those bookings were generated. Using data from Hotel ICON and breaking down the booking window in segments, they modeled the bookings-on-hand and time, as a quadratic function. They identified specific booking trends within each segment that can be a good indicator of the final bookings. This underscores the importance of identifying the booking patterns, as this thesis aims to do.

Finally, the industry standard is to employ a combined model (Webb et al., 2020). Usually this entails using a weighted average of the historic and advanced models to forecast; usually as the departure date approaches greater weight is assigned to the advanced model and vice versa. Ben-Akiva et al. (1987), as quoted in L. Weatherford and Kimes (2003), compared the performance of a historic

model using time series, an advanced model using regression and a combined model, to conclude that the combined model outperformed the rest.

De Sá (1987) also developed a combined model using multiple linear regression. The model assumed a causality between bookings-on-hand, bookings-to-come and used also other seasonality variables like week-of-the-year and day-of-the-week. The results showed that the combined model worked better than the historic, ARIMA model. This study found that bookings-on-hand do not exhibit good explanatory power for forecasting final bookings, which is in line with Tse and Poon (2015).

With the goal of increasing accuracy, various studies have also tried to incorporate new data in their forecasts. Pan et al. (2012) investigated the effectiveness of incorporating search engine data, specifically real-time, high-frequency data from Google Trends. By using ARMA (AutoRegressive Moving Average) and ARMAX (AutoRegressive Moving Average with Exogenous inputs) models, they incorporated data from five tourist-related queries to predict hotel demand. Their findings showed that all ARMA models were consistently outperformed by their ARMAX counterparts, demonstrating the explanatory value of search engine data in forecasting, even with the relatively low number of queries used. Subsequent studies, such as Li et al. (2019), have attempted to incorporate more query data compared to Pan et al. (2012), by developing a query evaluation system that determined the importance of the queries based on their correlation with past booking patterns, demonstrating that these queries have explanatory power in forecasting. On a similar note, Schwartz (2006) incorporated the forecasts of a hotel's direct competitors in their modeling. By using these competitive sets and a simple linear regression model in both a simulation scenario analysis and a field study, they showed a lower forecasting error in their predictive model of hotel occupancy, proving the value of leveraging market knowledge in forecasting. While these studies have been done for demand forecasting they can also be applied in the context of booking pattern forecasting.

Another consideration for forecasting according to L. Weatherford and Kimes (2003), is the choice of an appropriate error measure. Makridakis et al. (1982) concluded that the performance of forecasting models can vary depending on the error metric used. Furthermore, Kouprioucha et al. (2014) compared the results of seventeen error measures when forecasting hotel reservations, highlighting their differences in performance. This study emphasized the importance of comprehending the limitations of these measures to avoid inaccuracies that could lead to suboptimal decisions, suggesting selecting the most appropriate measures based on predetermined objectives.

Therefore, given the variability in model performance and potential error measures in forecasting, this thesis will consider a range of models and error metrics to forecast booking patterns, in line with Fildes and Lusk (1984) who recommended exploring multiple methods in forecasting.

2.2 Machine Learning in Forecasting

In the last decade, there has been a growing interest in the potential of Machine Learning in forecasting, as these models have been demonstrating their capabilities in the field (Ahmed et al., 2010). Impressive progress has been made through an analysis of the strengths and limitations of these models, often resulting in the development of novel, specialized models. For instance L. Huang and Zheng (2021), developed a novel forecasting model using Deep Learning and the Bayesian Optimization Algorithm.

Machine learning is applicable to Sunweb's case, as Zhang (2019) highlighted its enhanced performance in forecasting tasks involving large volumes of data. These models excel at capturing complex and non-parametric relationships in historic booking patterns, including trends and seasonality. Zhang's study compared various models, including Tree Models, Random Forest (RF) and Support Vector Machines (SVM), in terms of variance, bias, and accuracy. The findings revealed that SVM outperformed all other models with long historical booking information, while RF outperformed the others with short historical booking information.

Another ML model comparison study by Masini et al. (2021), compared the performance of both penalized Linear models (Ridge, Lasso, Elastic Net and variations) and Non-Linear models (Neural Networks and Tree-based models) for time-series forecasting. This research showed that non-linear ML models combined with large data sets can be extremely useful for forecasting and they performed well even in datasets with unprecedented changes, such as the Covid-19 pandemic. This ability to capture complex patterns, can be particularly useful for Sunweb, especially when trying to forecast non-consistent and fluctuating booking patterns.

Research has also focused on using Neural Networks and in particular Multi-Layered Perceptrons (MLP) as a forecasting technique (Webb et al., 2020; L. R. Weatherford et al., 2003). Neural networks with their non-parametric form can capture non-linear relationships, distinguishing them from the traditional time series models (L. R. Weatherford et al., 2003). L. R. Weatherford et al. (2003) showed that the MLP model achieved higher accuracy compared to traditional models like smoothing and regression, when forecasting weekly airline reservations. Similarly, Webb et al. (2020) showed that MLP and by extension neural network-based methods can be useful in forecasting in an advanced booking environment.

In the literature there has been some contrast in the application of Machine Learning in forecasting. Zhang (2019) noted that despite their enhanced performance, ML models are complicated to build, challenging to interpret and computationally demanding. Another concern in the application of ML in forecasting, arises from their relative performance compared to other models, which has

shown varying perspectives in the literature. On the one hand, Makridakis et al. (2020) demonstrated that not all ML models can consistently outperform traditional statistical models. On the other hand, Pereira and Cerqueira (2021) when comparing 22 ML models and traditional models, concluded that the ML models outperformed the traditional forecasting models, achieving up to a 45% reduction in the root mean squared error for a 14-day forecast horizon. According to Cerqueira et al. (2019), one major consideration for this variation in performance is the sample size. Their results demonstrated that ML models are competitive with traditional forecasting models when the training sample size is sufficiently large, so a number of ML models can be therefore applied in this thesis.

2.3 Clustering

Ma et al. (2014) and Schwartz (2008) argued that the distribution of historical booking patterns influences the forecasts of future bookings, suggesting a potential correlation between them. Various studies have shown enhanced performance in forecasting, when the model is trained on clustered data, demonstrating that by leveraging this information, the accuracy of booking forecasts can be improved (Ezugwu et al., 2022; Jain, 2010; Ma et al., 2014; Viverit et al., 2023).

Viverit et al. (2023) showed the effectiveness of using similar booking patterns for short-term forecasting and argued that understanding the behavior associated with them is fundamental for better forecasting. Their research showed that by forecasting at cluster-level instead of an aggregated-level, better forecasts can be made since the models are trained with booking patterns that exhibit similar behavior, making it easier to capture the underlying trends.

Clustering techniques fall under the umbrella of unsupervised machine learning, with the goal of detecting patterns in the data and making groupings without having predefined labels. According to Jain (2010), clustering has three main objectives: finding the underlying structure of data, identifying the degree of similarity by naturally grouping them, and compressing data by organizing and summarizing them through cluster representatives.

For instance, when applied to booking patterns, clustering can show that some holiday packages may showcase earlier reservations, while others have a more evenly distributed pattern or even experience peaks in last-minute bookings (analogous to results presented by Viverit et al., 2023). This distinction is critical for Sunweb, as understanding the timing of bookings for each departure week and destination can offer valuable insights and can further assist in accurately forecasting booking patterns. Additionally, certain holiday packages have only a small number of passengers, leading to volatile and unreliable booking patterns. According to Hyndman, Kostenko, et al. (2007) these fluctuations in a dataset can produce inaccurate predictions. By using clustering, Sunweb can leverage information

from other booking patterns that belong in the same cluster, and therefore behave similarly, to mitigate these issues and produce more accurate results.

Methods to deal with time-series clustering can be subcategorized in three main subcategories: based on curve similarity measures, indirectly with feature extraction from the series or indirectly with models built from raw data (Liao, 2005).

In terms of the similarity measures, Schwartz and Hiemstra (1997) employed a curve (dis)similarity method to build a forecasting model, leveraging the patterns shown by the booking curves. Their method concerned an advanced booking model and compared the available part of each incomplete curve with a historical curve up to that point in time, using the bookings-on-hand. By using ten different booking windows ranging from 1 to 99 days in advance, they incorporated past booking patterns into forecasting hotel bookings. Their model managed to forecast future bookings with higher accuracy when compared with traditional time series models and polynomial regression.

Another method based on curve similarity measures is the Time Series k-means (TS k-means) algorithm, which was first introduced by X. Huang et al. (2016). TS k-means, segments the entire sequence into distinct smooth subspaces and clusters time series based on these subspaces, rather than the entire sequence in chronological order, as proposed by Schwartz and Hiemstra (1997). TS k-means identifies the subspaces that exhibit distinct patterns and allocates weights to each based on their significance in the clustering process. This process prioritizes identifying the most discriminative temporal features in the series and assigns higher weights to certain timestamps that exhibit smaller variability, thereby ensuring a better split of the dataset.

The second category concerns the extraction of features and structural characteristics from the time series (Wang et al., 2006; Bandara et al., 2020). In this clustering method, time series are not clustered using a distance metric, but using features extracted from them such as their mean, variance, peak strength and strength of spikeiness. Numerous studies investigated the most important potential features to best compare time series (Fulcher and Jones, 2014; Wang et al., 2006). This method can be seen as a dimensionality reduction process for time series data and is predominantly used for its capability to manage high-dimensional (long time-series) data, accommodating missing values or dealing with time series of varying lengths (Bandara et al., 2020).

The third category is model-based methods, that assume a model for each one of the clusters and attempt to best fit the data into this assumption. This approach tends to suffer from scalability challenges and may exhibit decreased performance (Aghabozorgi et al., 2015), so it will not be employed in this thesis.

Clustering can be highly effective for booking pattern forecasting, but defining the ideal clustering technique can be challenging. This can be attributed to the various parameters involved, including

feature selection, defining similarity measures, randomization and deciding on the optimal number of clusters to ensure distinct and interpretable clusters (Jain, 2010; Kodinariya, Makwana, et al., 2013). An ideal cluster can be defined as a set of observations that exhibit internal cohesion and external isolation, meaning a low within-cluster variance and as high variance as possible for the neighboring clusters to ensure clear separation (Amigó et al., 2009).

2.4 Conclusion and proposed methods

To conclude this literature review, this thesis aims to contribute to the domain of booking pattern forecasting in two primary ways. Firstly, by evaluating the benefits of incorporating clustering in the model training. Two clustering methods are chosen; based on similarity measures and using feature extraction. For similarity measures clustering, the TS k-means algorithm by X. Huang et al. (2016) is employed, as it was specifically designed for time series analysis, can support multiple distance metrics and thus can result in more meaningful splits of the data. The similarity method proposed by Schwartz and Hiemstra (1997) can not be implemented in this context, since a historical model needs to be built. The feature extraction method is also deemed useful, particularly considering that it can reduce the high-dimensionality of the given dataset, making it computationally efficient and highly scalable for implementation by Sunweb. Moreover, a lot of research has focused on identifying the most informative features to include (Fulcher and Jones, 2014; Wang et al., 2006), which can be useful.

Secondly, this thesis aims to evaluate different forecasting models, by comparing the performance of (penalized) linear models with non-linear models, similar to Masini et al. (2021). This comparison aims to assess whether these models can effectively capture the various patterns present in booking data, encompassing both a simple linear and a more complex non-linear approach. Linear Regression based models are chosen for their computational efficiency and simplicity and Regression Tree models are selected for their hyperparameter tuning ability, easier interpretation compared to other ”black box” models like random forests, along with their flexibility in capturing non-linear trends (Gordon et al., 1984).

3. Data

3.1 Case background

Sunweb Group is one of the largest holiday providers in Europe that offers complete holiday packages to their customers. Currently they serve more than 1.2 million customers annually (Sunweb Group, 2023). Sunweb Group consists of six different brands but for this thesis, data from Sunweb Group's most widely recognized brand, Sunweb, was utilized. Sunweb focuses on providing holiday packages primarily to Northern European travelers, with popular destinations in Southern Europe and North Africa, notably Greece and Spain (Sunweb Group, 2023).

Sunweb prepurchases its package capacity well in advance, to ensure the best deals. Therefore, it is essential to have accurate forecasts of the distribution of the expected bookings and how those accumulate over time for a certain holiday package. This can enable Sunweb to respond to variations in performance of holiday packages. This can entail dynamic pricing, such as lowering prices to mitigate losses, marketing planning adjustments, like boosting marketing expenditure if performance is below expectations, capacity screening, and cash flow projections.

For this thesis, various historic forecasting models were built to generate predictions for the booking patterns with a predetermined booking window of 104 weeks (equivalent to 2 years), to serve as benchmarks for the company. The objective is to forecast accurately the percentage of bookings that will be made for each unique holiday package (defined as a unique combination of destination and departure) on a weekly basis. For example, predicting that 5% of the total bookings of a holiday package will be made one week prior to the departure week.

A percentage-based methodology was adopted to analyze weekly booking patterns relative to the total number of booked passengers of each holiday package. This way, comparisons of a holiday package's booking pattern with capacity of 100 can be compared with one of 1000 passengers. The aim was to normalize the data, especially considering the variability in the total number of passengers for each package. Moreover, the same 104-week booking window was employed for all packages, allowing for comparisons of booking patterns across all departure weeks. This consistent timeframe enables comparisons between bookings made even months apart, focusing on the relative timing of bookings compared to their respective departure week (i.e., number of weeks in advance of departure) within the fixed window. Overall, this methodology allows for comparisons between holiday packages, independent of their departure week and total number of passengers, offering better insights into booking patterns over time.

3.2 Dataset used

The dataset used in this analysis consists of cross-sectional data indexed by every holiday package's departure (week and year). For each departure, the dataset includes information on the *destination airport*, *destination country*, and the number of *passengers* who booked a holiday package during a specific *booking week*. Additionally, for the *booking weeks* present in the dataset, the corresponding *revenue* and *margin* generated by Sunweb is captured. It should be noted, that the dataset does not contain information for every possible *booking week* corresponding to a specific *departure* for every holiday package, but in case it is missing, it can be inferred that no passengers booked during that week.

The data consists of organized holiday packages originating from the Netherlands, ensuring that national culture does not affect the booking behavior, and targeting 63 different airports across 15 different countries. In total, 355 booking weeks (Note: 2015 was a leap year) from the 3rd week of 2013 until the 44th week of 2019 and 209 departure weeks from the 44th week of 2015 until the 44th week of 2019 are taken into account. It is important to note that the booking week always precedes or coincides with the departure week. The training dataset was chosen as all the booking patterns with departures spanning from the 44th week of 2015 to the 44th week of 2018, while the test dataset encompasses the booking patterns departing from the 45th week of 2018 to the 44th week of 2019.

As mentioned, the data provided contains information regarding the revenue and margin (defined as the revenue minus holiday package costs); but they were not used in this analysis for two reasons. Firstly, Sunweb's algorithms change the pricing every day and therefore the weekly aggregation does not reflect the daily price fluctuations. Secondly, when missing weekly booking information, there is no record of the pricing of that booking week and therefore no information can be inferred regarding the revenue and margin. This thesis expects that pricing patterns will show consistent yearly trends, influenced by how far in advance the booking is made and consistent seasonal variations (approach aligns with Goia et al., 2010).

As stated, this thesis focuses on making accurate forecasts for the booking pattern of each holiday package. However, some holiday packages have fewer total passengers, leading to significant inconsistencies and variations in their year-to-year booking patterns and therefore raising concerns about data reliability. This variability can affect the forecasting accuracy of a model, as these patterns can be more challenging to predict since they are heavily affected by small fluctuations in weekly bookings. So Sunweb, prioritizes accurately forecasting bookings of destinations with more passengers. Small variations in bookings do not affect them significantly, making them more consistent and better representative of actual booking patterns. To address this issue, this thesis will develop a

number of models, prioritizing the forecasting of booking patterns for high-volume holiday packages with higher accuracy. Furthermore, the assessment of all models will also include the utilization of error metrics that focus on measuring the errors of holiday packages with higher total passengers. For the purpose of this thesis, a low-volume package is defined as one with fewer than 50 total passengers. It is important to note that a holiday package may be classified as low-volume in some years and not in others. In our dataset, based on this criterion, we classified 4787 holiday packages as high-volume and 1895 as low-volume.

To construct the dataset to maximize predictive power and ensure computational efficiency, the data was organized with each unique holiday package serving as the index. As a result, each row in the dataset represents a holiday package, displaying its booking pattern as a list of 104 elements. Importantly, since the data did not contain information for every possible booking week of every holiday package, it was inferred that if there was no data for a particular combination of *destination* (represented by the unique destination airport), *departure* (represented by unique departure year and week) and *bookingweek* within the 104-week forecasting horizon, it meant zero bookings were made for that period. For our explanatory variables, additional features were introduced for each holiday package, such as the total passengers and the low/high volume category based on the pre-set limit of 50 passengers, while revenue and margin variables were not used. The booking patterns were converted to percentages relatively to the total passengers of that holiday package, to enable comparisons between different departures.

To highlight the problems associated with the high and low volume booking patterns, the normalized accumulated bookings of two holiday packages were plotted in *Figure 3.1* as a function of the number of weeks in advance, each for four different years.

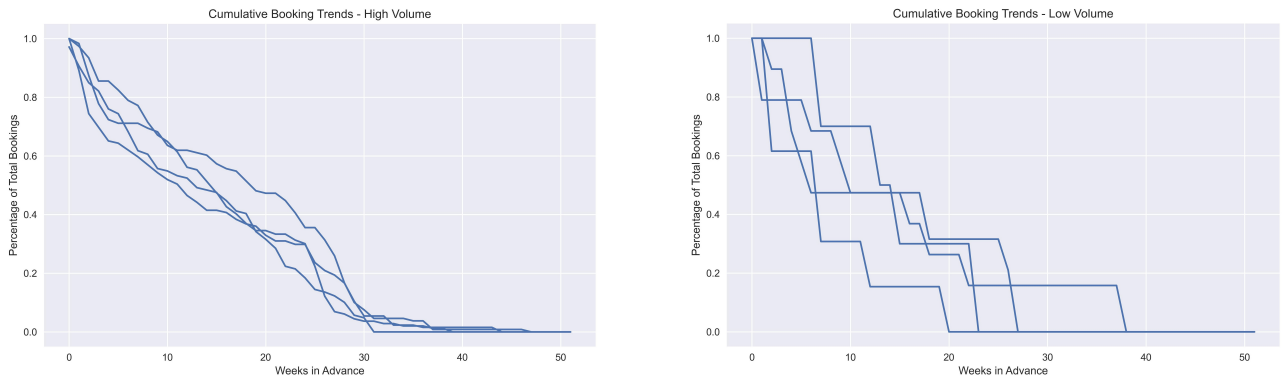


Figure 3.1: Booking patterns for high and low-volume holiday packages

Note: The figure was adjusted to only 52 weeks, but generally, bookings are made up to 102 weeks in advance.

Clearly the two graphs exhibit a different behavior; in the low-volume case (depicted on the right) there are big jumps that do not exhibit consistent yearly patterns. There are differences approximately at 50% in total capacity reached observed 12 weeks before departure in this instance. On the other hand, the plotted patterns in the high volume case are consistently closer and exhibit less variation.

Preliminary analysis has indicated the presence of booking trends; the booking process is influenced by diverse trends associated with two key variables: the booking week and the number of weeks in advance of departure. For the number of *weeks in advance* the booking was made, it was observed for example, that sometimes there were spikes in bookings during the final three weeks before the departure date, indicative of last-minute booking tendencies; sometimes bookings started to accumulate at different times for a holiday package, or even slowed down from time to time. Also, for the *booking week* variable, it was observed that certain booking weeks in a year showcased higher total bookings compared to others where bookings were lower. These two variables will be the primary variables in our modeling. An example is given: if the *DepartureYearWeek* is given as 2018_ 42 (meaning the 42nd week of 2018) and the *BookingYearWeek* is 2018_32 then the variable *"BookingWeek"* will be equal to 32 and *"Weeks in advance"* will be equal to 10.

4. Methodology

The aim of this thesis is to assess the performance of forecasting models in comparison to ensembles that are also trained on clustered booking patterns. The model evaluation will focus both on the forecasting errors on the overall dataset and also on errors of holiday packages with higher total passengers.

This thesis hypothesizes that clustering can improve forecasting accuracy by training on more consistent booking patterns, making it easier to identify and capture distinct trends. This can also potentially help overcome the volatility of the booking patterns of low-volume packages. By leveraging information from clusters, forecasts can be improved by transferring information from similar booking patters that belong in the same cluster, without the computational complexity of developing numerous flight-specific models, noted by Lee (1990). Additionally, clustering can provide valuable insights into booking trends and understanding how these patterns vary across destination countries and departure weeks.

4.1 General Models

4.1.1 Benchmark Average Model

All the models will be compared to a "naive" benchmark model (*model 4.1.1.1*), referred to as *Average Model* and calculated by averaging past booking patterns for each of the 52 departure weeks. Preliminary analysis has shown that the booking patterns are in general consistent across all destinations, given a distinct departure week. Therefore for each departure week, the forecasted percentage of total passengers in booking week *i*, is the average percentage made in the same booking week in previous years, for all the holiday packages. This will result in 52 different booking forecasts, each with a booking window of 2 years. The mathematical formulation is as follows: $y_{i,j,k,d}$ is the percentage of total passengers who booked in booking week *i* for a specific departure week *j*, departure year *k* and destination *d*. Therefore, for each departure week *j* the percentage of bookings made in booking week *i* are defined as the average of all the percentage of bookings of *T* years for all the available destinations *D* :

$$y_{i,j,k,d} = \frac{1}{N} \sum_{t=1}^{T} \sum_{d=1}^{D} y_{i,j,k-t,d} \qquad \text{(equation 4.1)}$$

where *N* is the total number of holiday packages that fit this criteria and *T* is the total number of available years in the training set.

Below there is a brief overview of the models and the estimation techniques that are employed in this thesis, namely linear regression-based models; penalized and not, and tree-based models. These models will be here referred to as general models, as they will be trained on all the booking patterns in the training set. As mentioned before, the variables used will be the *booking week* and the number of *weeks in advance* of departure. These will be incorporated in the form of dummy variables. Preliminary analysis indicated that if the departure week is farther away, weekly bookings are lower. This trend is particularly notable approximately a year before departure, where all bookings weeks show a similar low passenger percentage, close to zero. These findings suggest a grouping for the variable *weeks in advance*. This approach will group certain weeks in advance together and will assign the same coefficient to these grouped weeks, reducing the number of dummy variables included and also potentially helping to smooth out any anomalies. This can also help increase the computational efficiency of the models by reducing the high dimensionality of the variables and therefore result in faster training. In order to decide on an appropriate grouping, *Figure 4.1* was made. The figure depicts the weekly number of passengers who booked in a certain week in advance of their respective departure, against the weeks in advance variable, for the training set. It also shows, with the horizontal plotted lines, the 25th, 50th, and 75th percentiles of the total passengers, illustrating the intervals for each percentile group, plotted with a different color.

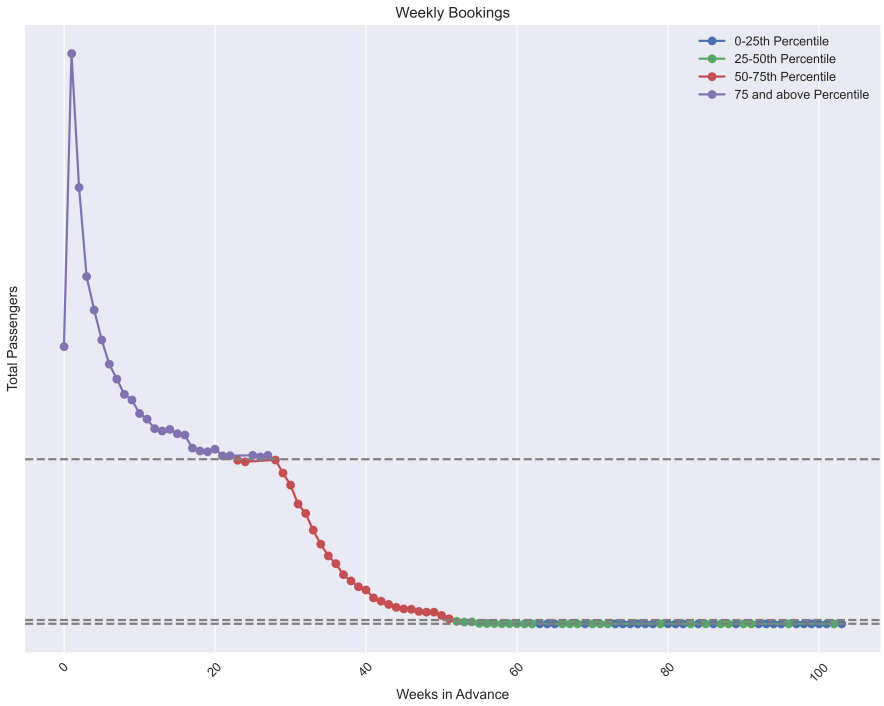


Figure 4.1: Number of bookings made per week in advance of departure

Note: As expected, the 0-25th and 25-50th percentile groups are very close, because many weeks in advance of departure, the total number of passengers booked is generally zero.

As it can be seen, there is a downward trend, meaning that in general the closer to the departure week, more weekly bookings are being made. For example, most bookings are being made one week before the departure. Using this figure, *Table 4.1* shows the grouping of the variable of weeks in advance and the number of dummy variables decided to be included for each percentile group. Some readjustments were made to ensure a sequential grouping and reducing the number of the variables.

Table 4.1: Groupings for Dummy Variables

Percentiles	**Weeks in advance**	**Grouping of Dummies**	**Total Dummy Variables**
75-100th	0-22, 25-27	One dummy each week	26
50-75th	23-24, 28-51	One dummy every two sequential weeks	13
25-50th	52-71	One dummy every four sequential weeks	5
0-25th	72-103	Grouping into one dummy	1

Another insight from *Figure 4.1* and initial analysis, is that there is a significant number of last-minute reservations. The last three booking weeks before departure (0, 1, 2, and 3 weeks in advance) account for about 21% of the total bookings. Even with the inclusion of grouped dummies for *weeks in advance* as described above, the initial models may not adequately capture these trends. For Sunweb, accurately forecasting last-minute peaks is crucial because they need to sell all their purchased capacity in an optimal price. For example, if a forecast fails to predict a last-minute peak, and Sunweb reduces prices to attract bookings that would have been made even without the price reduction, Sunweb could lose potential revenue. To address this, the dummy variables for the last three booking weeks before departure were multiplied by a factor of 2. By doubling the value for these booking weeks, the models aim to place more emphasis on accurately predicting last-minute bookings by increasing their influence on the model.

Therefore, every model will have a total of 149 input variables: 45 for the grouped weeks in advance (denoted as c_j), and 104 variables for each of the booking weeks (denoted as w_k), since our forecasting horizon is 2 years. It is important to note that 104 booking weeks were chosen instead of 52 to capture the significance of the booking week being either less or more than 52 weeks before departure. For example, if a booking was made in the 33rd week of the year and another booking was made in 33rd week of the previous year (more than a year in advance), the dummy variable will be different. This choice was made, since the analogous graph was not smooth and the percentiles calculated, gave non-sequential groupings, which are harder to understand.

The dependent variable is the forecasted percentage of total passengers for each holiday package (unique destination and departure), as before $y_{i,j,k,d}$. For simplicity in the following formulations it will be denoted simply as y_m, where *m* encompasses all the aforementioned indexes. For all the Linear Regression-based models, as seen below in subsections *4.1.2*, *4.1.3* and *4.1.4*, the general equation is given by:

$$y_m = \beta_1 c_1 + \cdots + \beta_{45} c_{45} + \gamma_1 w_1 + \cdots + \gamma_{104} w_{104} + \epsilon \qquad \text{(equation 4.2)}$$

This equation will be solved with multiple *Estimation Techniques* of different models, as shown below.

4.1.2 Multiple Linear Regression

Linear Regression is a statistical technique that models the relationship between a dependent variable and two or more independent variables by fitting a linear equation to the observed data, with the goal of minimizing the sum of squared errors (Weisberg, 2005). The estimation technique that will be used is Ordinary Least Squares (OLS), whose objective is to minimize the sum of squared errors (SSE) between the true value y_m and the predicted value $\hat{y}_m$ denoted here as J_{OLS}:

$$\mathrm{J_{OLS}} = \sum_{m=1}^{M} (y_m - \hat{y}_m)^2 \qquad \text{(equation 4.3)}$$

This model was trained under two scenarios; one where all booking patterns were included, namely *model 4.1.2.1* referred to as *OLS*, and another where the linear regression was trained only on booking patterns of high-volume packages, namely *model 4.1.2.2* referred to as *OLS (high volume)*. The reason behind the choice of *OLS (high volume)* was that high-volume booking patterns exhibit more consistent and reliable trends compared to the low volume ones.

4.1.3 Weighted Linear Regression

Weighted linear regression is an extension of the multiple linear regression model, that allows for heteroscedasticity of the dependent variable, in which each observation is assigned a weight (w_m) that reflects the importance or reliability of the observation (Thompson, 1982). Higher weights are assigned to observations that need to be fitted more accurately. The estimation technique used is Weighted Least Squares (WLS), whose objective is to minimize the weighted sum of squared errors denoted as J_{WLS}, which is similar to the estimation technique of OLS but multiplied by a weight that reflects the relative importance of each observation:

$$\mathrm{J_{WLS}} = \sum_{m=1}^{M} w_m (y_m - \hat{y}_m)^2 \qquad \text{(equation 4.4)}$$

Two models were trained to optimize the weight selection; Firstly weights that were proportional to the total number of passengers on each holiday package, namely *model 4.1.3.1*, referred to as *WLS (proportional)* . This means that for each holiday package, the weight assigned was calculated as the ratio of the total number of passengers for that package to the total number of passengers across all packages. Secondly, *model 4.1.3.2*, referred to as *WLS (doubled)*, the weights used were doubled when the package is considered high-volume compared to low-volume. Both of these models aim to put more emphasis on correctly fitting the reliable booking patterns of high-volume holiday packages.

4.1.4 Ridge Regression

The Ridge regression is a penalized regression model with a penalty term proportional to the square of the coefficients (commonly referred to as l2 regularization) to prevent overfitting. Ridge regression computes an easy analytic solution, as relevant predictor variables are "shrunk" towards zero (Masini et al., 2021). The estimation technique used is Ridge Regression, which introduces a regularization term to the OLS objective to prevent overfitting, denoted here as J_{Ridge}. The objective is to minimize the sum of squared errors (SSE) along with a penalty proportional to the sum of the squared coefficients. The regularization parameter λ controls the degree of shrinkage.

$$\text{J}_{\text{Ridge}} = \sum_{m=1}^{M}(y_m - \hat{y}_m)^2 + \lambda(\sum_{j=1}^{J}\beta_j^2 + \sum_{k=1}^{K}\gamma_k^2) \qquad \text{(equation 4.5)}$$

In this *model 4.1.4.1*, referred to as *Ridge*, ten equally-spaced values of λ will be assessed in the range from 2 (some regularization) to 20 (stronger regularization). These values were chosen in an attempt to ensure robustness, in a wide range of uniformly distributed regularization parameters with computational efficiency, as the limit was set to a relatively efficient upper limit of 20.

4.1.5 Regression Tree

A Regression Tree is a non-parametric model that fits the data using a non-linear function by partitioning the data space, using nodes. It has a hierarchical tree structure. The CART algorithm (Lewis, 2000) was employed that greedily selects splits based on the predictor variables to minimize the sum of squared errors within each node (Gordon et al., 1984). To make predictions, the tree is followed from the top down to its endpoints, called leaf nodes. The predicted value for each case is then the average of the target variable values in the end node that it ends up in. The objective is to minimize the MSE within each subset, created by the splits. The MSE for a node t is formulated below, where N_t is the number of observations in node t, y_m is the observed value and $\bar{y}_t$ is the mean of the values in node t:

$$\text{MSE}_t = \frac{1}{N_t}\sum_{m\in t}(y_m - \bar{y}_t)^2 \qquad \text{(equation 4.6)}$$

The overall objective of the regression tree is to minimize the total MSE across all nodes.

$$\text{Total MSE} = \sum_{t\in T}\text{MSE}_t \qquad \text{(equation 4.7)}$$

In this *model 4.1.5.1*, namely *Regression Tree*, the parameters used for tuning are as follows: the maximum depth of the tree (max_depth), set to the range of 7 equally spaced values from 20 to 50; the minimum number of samples required to split a node (min_samples_split), set to either 5 or 10; the minimum number of samples in a leaf node (min_samples_leaf), set to either 10,15,20 or 25; and the maximum number of features to consider for each split (max_features), set to either the square root or half of the total features (dummy variables included). These specific parameters were chosen to balance the tree complexity, node splitting conditions, training time and feature selection, aiming to optimize predictive performance while controlling for overfitting.

For all the models above, the hyperparameter tuning was conducted in the training dataset using gridsearch and five-fold cross-validation (GridSearchCV). Instead of using a standard scorer function for hyperparameter tuning, after experimenting with multiple alternatives, a custom scorer was created to identify the combination of parameters that minimizes the sum of the Mean Squared Error (MSE) and Mean Absolute Error (MAE).

As mentioned, the training dataset consists of all the booking patterns of the departure weeks between the 44th week of 2015 until the 44th week of 2018. However, *Figure 3.1* illustrated the issues associated with the consistency of low-volume booking patterns and therefore to optimize the model training and to capture the stronger booking patterns, it was decided to exclude some booking patterns from the training dataset. In order to choose the holiday packages whose patterns would be excluded from the training process (both for the general models and clustering later on), a similarity approach with a threshold was employed. For every holiday package in a certain departure week, the cosine similarity between that pattern and the average pattern as per *equation 4.1* for that departure week was calculated. This is to assess how closely that booking pattern aligns with the ”expected” booking behavior of that departure week. Then, the average similarity of those patterns was calculated for each distinct number of total passengers, as shown in *Figure 4.2*.

Figure 4.2: Mean Similarity score of booking patterns for total number of passengers

It can be seen from the figure that generally there is an upward trend; meaning that the booking pattern is more consistent with booking pattern produced by the average model if the holiday package has more total passengers. It was decided that observations with similarity scores below the threshold of 0.5 , are less representative of the booking patterns and will be excluded from the training dataset. This includes all holiday packages where the total number of passengers is less or equal to 26, and 354 (Note: in this Figure the total number of passengers was hidden due to confidentiality).

4.2 Clustering and Ensembles

For the ensemble models, with the aim to assess the benefits of including the cluster forecasts, a four-step approach was applied:

- Compute the past average booking patterns per holiday package (see *equation 4.8* below)
- Cluster those booking patterns using different methods.
- Train a forecasting model on each individual cluster.
- Build an ensemble model using the weighted cluster forecast and the general model forecast.

Firstly, for each holiday package the historical average curves were calculated. These curves differ from the ones calculated from *equation 4.1*, because these are based on the unique holiday packages (defined as a unique combination of destination and departure week) and not just unique departure week. The mathematical formulation is similar as before with $y_{i,j,k,d}$ representing the forecasted percentage of total passengers in booking week *i* for a specific departure week *j*, departure year *k* and destination *d*. Therefore, for each unique holiday package, meaning unique values of departure week *j* and destination *d*, the bookings made in booking week *i* are defined as the average of all the booking percentages of the *T* years in the training set:

$$y_{i,j,k,d} = \frac{1}{T} \sum_{t=1}^{T} y_{i,j,k-t,d} \qquad \text{(equation 4.8)}$$

The reason behind utilizing the average booking patterns per holiday package in the clustering was that in general they showcased a good overall predicting performance of future bookings. Additionally, this approach reduces the number inputs of booking patterns for clustering, as opposed to clustering all the available booking patterns, and also accommodates numerous combinations of destinations and departure weeks. Moreover, averaging the past booking patterns can smooth out anomalies and therefore reduce the variance of the forecasts, leading to more robust forecasting outcomes.

Secondly, the patterns were used to cluster, based on different methods depicted below:

4.2.1 Clustering using TimeSeries k-means

The Time Series k-means (TS k-means) algorithm was employed, since it is designed specifically for time series data and offers the option to use flexible distance measures, allowing it to better capture patterns. In this context, the Euclidean distance and Dynamic Time Warping (DTW) were used as distance metrics. DTW is a distance metric for aligning two time-dependent sequences by allowing for nonlinear, point-by-point warping that can accommodate groupings of booking behaviors that may not perfectly align week-by-week (Müller, 2007).

This algorithm was applied for two distinct groupings. Firstly, *Clustering 4.2.1.1*, referred to as *TS k-means*, was performed using the average booking patterns per holiday package of previous years as input, each formatted as a series with 104 elements. Secondly, for *Clustering 4.2.1.2*, referred to as *TS k-means (Aggregated)*, the

booking weeks in advance of the patterns were aggregated in pairs. For example, the passenger percentages of week 0 and 1 before departure were aggregated together. This aggregation was performed using the formula: $y_n = y_m + y_{m+1}$ for $m = 0, 2, ...$ This resulted in a reduction of the total series from 104 elements to 52. The rationale behind this approach is to improve the grouping of holiday packages that may experience some week by week variation.

4.2.2. Clustering using Curve Characteristics

For this clustering a number of features extracted from the average booking patterns were used to cluster, using the k-means algorithm. The k-means is a simple, computationally inexpensive, partitioning clustering algorithm that aims to minimize the within-cluster sum of squares (Ikotun et al., 2023). The features utilized in this thesis were collected from various papers and can be found below. Note that after extracting the features, they were standardized by removing the mean and scaling to unit variance; to ensure that all features will contribute equally to the clustering. *Clustering 4.2.2.1*, referred to as *Feature Clustering*, is expected to significantly reduce the computational time compared to any clustering using the TS k-means algorithm.

Feature	**Description**
First order autocorrelation	Autocorrelation with 1 lag
Trend	Calculated as the slope coefficient when fitting an OLS Regression with a constant using the booking patterns and the ”weeks in advance” as variables per holiday package
Skewness	Skewness of the booking pattern
Kurtosis	Kurtosis of the booking pattern
Mean	Mean value of the booking pattern
Variance	Variability of the booking pattern
Trough Depth	Minimum value of the booking pattern (this was set by the researcher to be non-zero)
Spikiness	Maximum difference between two consecutive observations
Level Shift	Maximum deviation of the data from its rolling mean with a window set to 3 weeks
Peak Frequency	Number of times the curve exceeded 0.15 (this was set by the researcher)
Peak Strength	Maximum value of the booking pattern
Seasonality Strength	Decomposing the series into trend, seasonality, and residual. Calculated as the ratio of the variance of seasonality over the summation of the variance of seasonality plus the variance of the residual

The assessment of the clustering was done using silhouette scores, the Davies-Bouldin index, and inertia graphs (Zuccarelli, 2021). The objective was to maximize the silhouette score, which measures how well-separated and cohesive the clusters are. *TS k-means* and *TS k-means (Aggregated)* clusterings were assessed to optimize the number of clusters (ranging from 2 to 6) and the choice of distance metric (either Euclidean or DTW). *Feature Clustering* was also evaluated in order to optimize the number of clusters (ranging from 2 to 6).

Thirdly, a forecasting model is trained on each individual cluster. Two models were selected: Multiple Linear Regression and a Regression Tree (descriptions can be found above in subsections 4.1.2 and 4.1.5). This selection enabled the inclusion of both a simple linear and a more complex non-linear approach, facilitating a more robust examination of various booking patterns. For the hyperparameter tuning of the regression trees the parameters used are the same as used for the general models.

After that, ensemble models were built. The fundamental idea behind ensemble learning is to train multiple base models, combine their predictions to account for uncertainties and errors and improve the forecast accuracy of the overall model (Wu & Levinson, 2021). Forecasts were generated using two models: one trained on the clustered data and another from the general model. These forecasts from both models were combined. The forecasts of the linear regression model trained on each cluster and the linear regression model trained on the entire dataset were combined, and the same process was performed for the Regression Trees. The combination is achieved through a weighted prediction. The weights are optimized on the training set to ensure optimal performance. The performance was evaluated using a combinaiton of some weights that sum up to 1 for both forecasts, with a constraint that the weight of the per-cluster model forecasts must be at least 0.5; as the aim of the ensemble modeling was to assess the benefits of including these per-cluster forecasts.

A graph of the overall methodology can be found below:

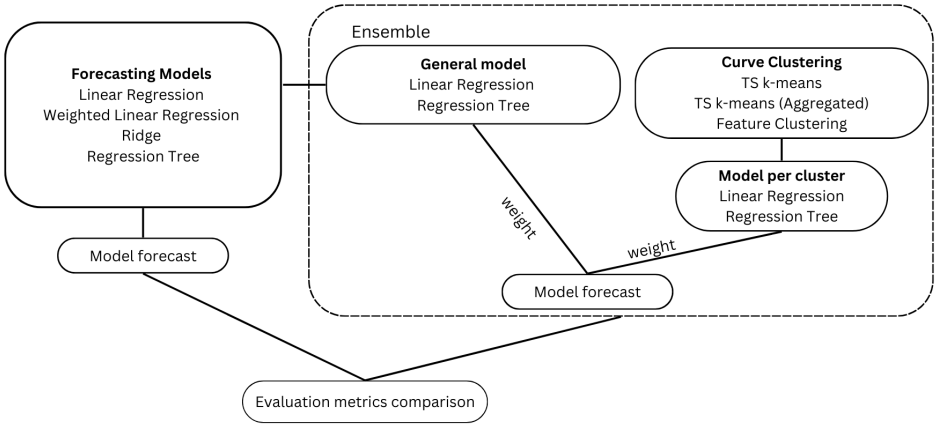


Figure 4.3: Methodology

4.3 Evaluation

Considering the findings of Kouprioucha et al. (2014) regarding the evaluation metrics, this study implemented the widely accepted scale-dependent measures of the Mean Average Error (MAE) and Mean Squared Error (MSE). As can be seen by *metric 4.3.1* and *metric 4.3.2*, the MAE measures the average magnitude of errors between the actual (y_m) and predicted values ($\hat{y}_m$) using an absolute value and the MSE on the other hand, uses their squared differences. For the purpose of this thesis, the Weighted MAE and MSE given by *metric 4.3.3* and *metric 4.3.4* were also selected, to properly adjust for the varying significance of holiday packages depending on their total passengers. The reason for this choice was to prioritize measuring the errors of the high-volume holiday packages, as they have more stable and reliable booking patterns. Forecasting low-volume booking patterns is relatively less critical because small changes in bookings can significantly alter the weekly percentage of expected bookings, whereas for high-volume packages, small changes have less impact on the booking pattern. Therefore, the weights w_m were calculated uniformly for each holiday package, defined as the ratio of the total number of passengers for that package to the total number of passengers across all packages. All the weights calculated, add up to 1 and higher weights are assigned to the holiday packages with higher total passengers. The formulation of all the error measures therefore, for this thesis are:

$$\text{MAE} = \frac{1}{M}\sum_{m=1}^{M}|y_m - \hat{y}_m| \qquad \text{(metric 4.3.1)}$$

$$\text{MSE} = \frac{1}{M}\sum_{m=1}^{M}(y_m - \hat{y}_m)^2 \qquad \text{(metric 4.3.2)}$$

$$\text{WMAE} = \frac{1}{M}\sum_{m=1}^{M}w_m|y_m - \hat{y}_m| \qquad \text{(metric 4.3.3)}$$

$$\text{WMSE} = \frac{1}{M}\sum_{m=1}^{M}w_m(y_m - \hat{y}_m)^2 \qquad \text{(metric 4.3.4)}$$

5. Results

5.1 Clustering Results

The hyperparameter tuning for the clustering, indicated the following optimal parameters: for the *TS k-means* three clusters were chosen with the DTW distance metric, for the *TS k-means (Aggregated)*, the chosen number of clusters was three using the Eucledian distance metric and for the *Feature Clustering* three clusters were also chosen. Having an equal number of clusters across all clustering methods was preferred, since this ensures easier comparison between the cluster labels assigned by the different methods. Also, only having two clusters may not allow a deeper understanding of the driving factors behind the clustering. Therefore, although the optimal number of clusters sometimes appeared to be two, it was decided to set the minimum number of clusters at three.

An example of the insights generated from the different clusters is shown on Figure 5.1, based on *TS k-means*. The medoid curves for each cluster, which are the booking patterns with the smallest average distance to all the other booking patterns in that cluster were plotted in a cumulative form for interpretability.

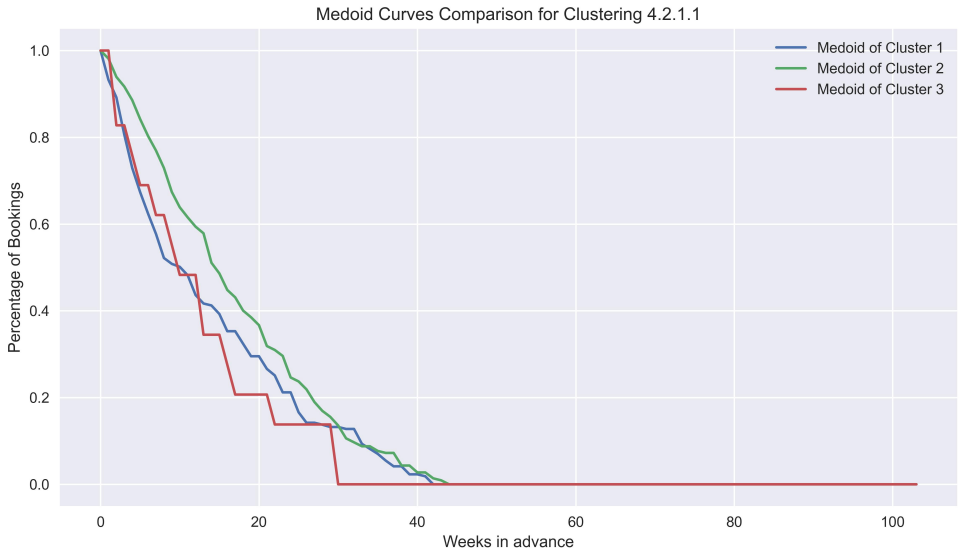


Figure 5.1: Medoid Curves of TS k-means Clustering

When comparing the medoid curves, it can be seen that they exhibit different behavior. In the first cluster, bookings typically begin to accumulate approximately 41 weeks before departure, generally accumulating steadily with a minor pause 25-30 weeks before departure and there are a lot of reservations made in the last ten weeks; namely approximately 50% of the reservations are being made the last 10 weeks before departure. This pattern suggests a strong tendency for closer-to-departure bookings and a relative lack of consistent bookings during the earlier period. In contrast, cluster 2 shows that reservations start to accumulate approximately the same time as in cluster 1, but with more consistent and gradual increase week by week than in cluster 1. This

pattern indicates the expectation of a consistent number of weekly bookings. For the third cluster on the other hand, reservations begin closer to departure than in the other 2 clusters; approximately 30 weeks prior, with significant jumps, a steep rise and as in cluster 1 a lot of relatively later reservations, illustrating inconsistent weekly patterns.

Analogous discussion can be done for the other two clustering methods, whose graphs can be found in *Appendix B*. In general, when comparing the medoid curves among the three clustering methods, the medoids of *TS k-means (Aggregated)* exhibit the most distinct booking patterns, while the medoid curves of the other two clustering methods are more similar to each other.

An important aspect to analyze for the clustering is how the booking patterns are distributed in the clusters and understand the underlying trends. For this, two characteristics of each holiday package will be explored; the *departure week* and the *destination country*. For instance, can it be said that the majority of holiday packages to Greece or departing between weeks 1-4 predominantly belong to a specific cluster? To analyze this, firstly it was decided to perform a grouping on the *departure week* to see which booking patterns exhibit similar behavior. Instead of grouping the departure weeks sequentially, assuming similar patterns within neighboring departure weeks, an unsupervised method of hierarchical clustering was preferred. Hierarchical clustering is useful in this case, because it begins by assigning each observation as its own cluster and then merges them, unveiling the hierarchical structure and relationships between them. The average curves as per equation 4.1 were utilized in order to uncover hierarchical relationships within the booking patterns between the 52 different departure weeks. The Dynamic Time Warping (DTW) distance was used as a distance metric and the Ward linkage method was selected, since it aims to minimize the variance during cluster formation, ensuring more cohesive grouping (Bismi, 2023). In order to maximize the silhouette scores for the hierarchical clustering and determine the optimal number of clusters, 8 clusters were selected as optimal.

In order to visualize this, the dendogram was plotted in Figure 5.2. The dendrogram visually shows how clusters are formed and merged in every iteration of the clustering. Also, the height of when the merger between clusters occurs, shows the (dis)similarity between the booking patterns. Booking patterns being merged at a lower height showcase greater similarity and the ones being merged at higher heights illustrate higher dissimilarity. In this case for example, in the first iteration departure week 25 and 24 illustrate more similarity in their booking patterns than week 3 and 4. This dendogram is particularly interesting, because it shows that initial groupings often occur between consecutive departure weeks, whereas later mergers involve departure weeks that are farther apart in sequence. For example the booking pattern of departure week 26 is being merged with the cluster of booking patterns containing departure weeks 31 and 32, and the same occurs for departure week 46 with 8 and 9. The corresponding groupings can be seen in the Table 5.1 below as well.

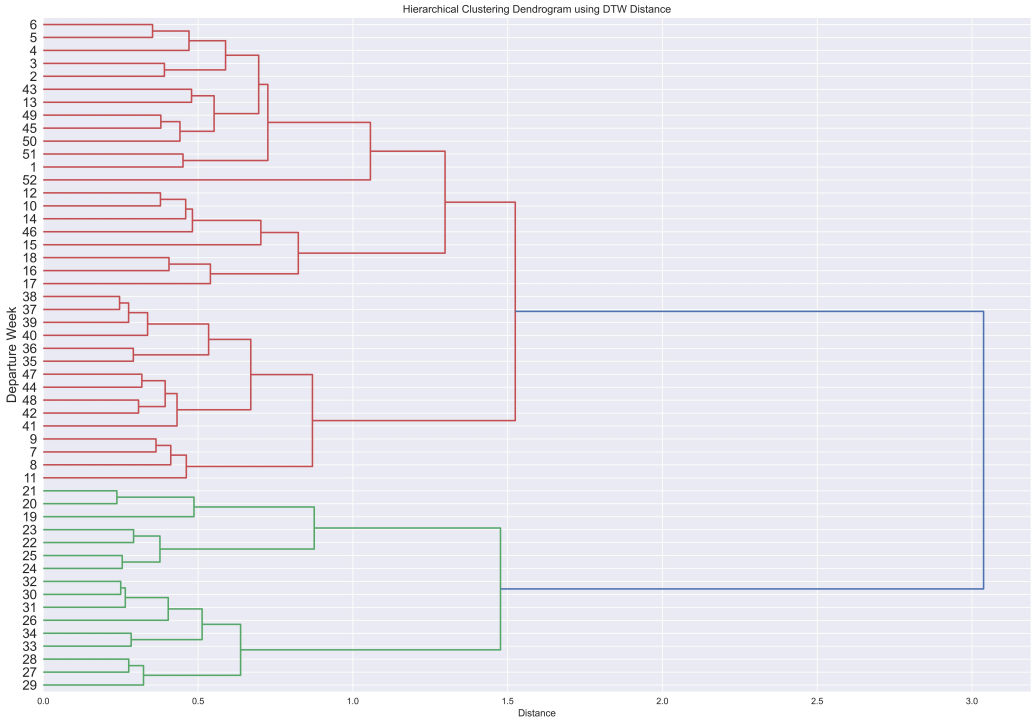


Figure 5.2: Dendogram of departure weeks

Table 5.1: Grouped Departure Weeks

Number of Group	Departure Weeks
1	29, 30, 31, 32, 33, 34
2	25, 26, 27, 28
3	20, 21, 22, 23, 24
4	35, 36, 37, 38, 39, 40, 41, 42
5	10, 11, 12, 13, 14, 15, 16, 46, 47, 48, 49
6	1, 51, 52
7	6, 7, 8, 9, 43, 44, 45
8	2, 3, 4, 5, 17, 18, 19, 50

From the table of grouped departure weeks above, it is evident that many groups consist of sequential departure weeks, but each with a different number. For example, departure weeks 1, 51 and 52 belong in the same group while the following sequential weeks namely 2,3,4 and 5 belong in another group. Also noteworthy, is the fact that a lot of groups contain departure weeks from two distinct sequential periods, for example in the 7th group of departure weeks, the departure weeks range from 6-9 and from 43-45. Further domain knowledge is required to understand what drives this grouping.

Continuing this analysis; the Figure 5.3 below provides insights into the relationship between the *destination country* and *grouped departure week* variable for each cluster assignment of *TS k-means*. Given the uneven distribution of the data, where certain popular destinations and departure weeks are more frequently represented,

the proportion of each variable in the cluster instead of the absolute values of each cluster were used. The figure illustrates the distribution of grouped departure weeks and countries within the clusters. For instance, approximately 70% of all holiday packages to Greece are categorized in cluster 2, while approximately 30% fall into cluster 1, and so forth. It is important to mention that also some destination countries, namely Gambia and the UAE fall completely into one cluster. Similar conclusions can be derived for the *grouped departure week*. For example, taking two departure week groups with sequential departure weeks, namely group 1 (departure weeks from 29-34) and group 2 (departure weeks from 25-28) it can be seen that for group 1 more booking patterns fall in cluster 2.

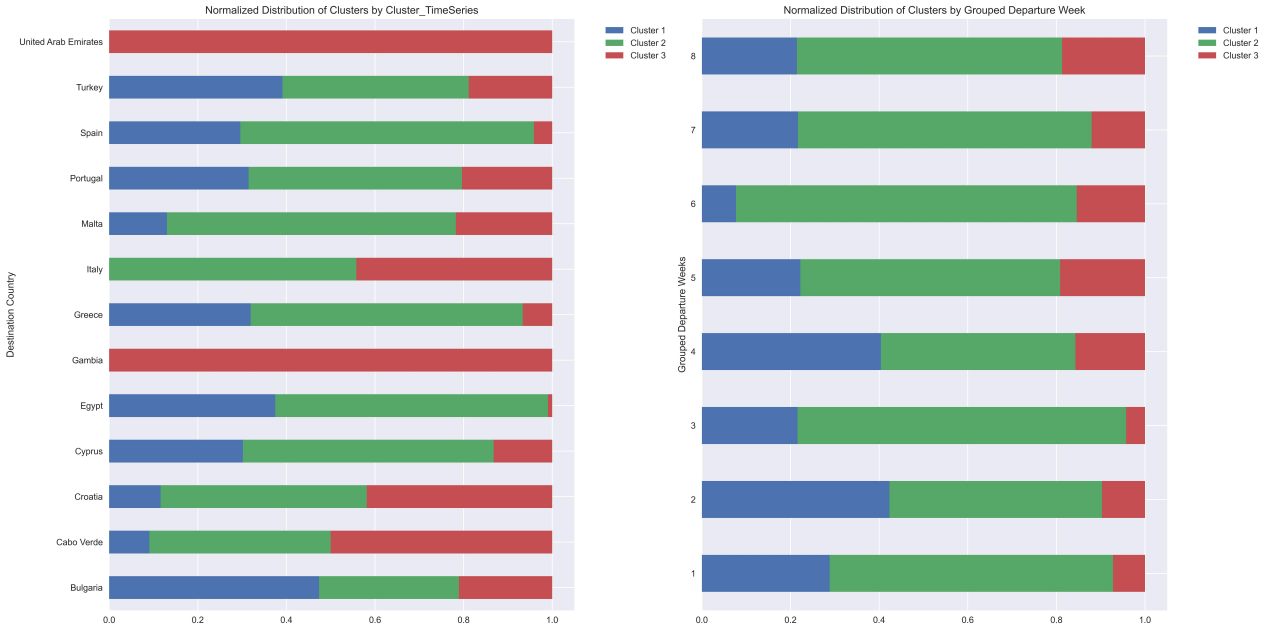


Figure 5.3: Analysis of destination and grouped departure week in clustering

To further understand the factors driving the cluster determination, the relative importance of the grouped departure week versus the destination country was examined. For this, the feature importances using a random forest analysis were computed, as they can showcase the importance of each variable in the cluster assignment, based on the number of higher order splits. In other words, they reveal which variable plays a more important role in the cluster labeling. The analysis revealed that for *TS k-means*, the destination country accounted for approximately 70% of the importance, while the grouped departure week contributed 30%. Similarly, for *Feature Clustering*, the destination country had the same predominant role at approximately 70%, suggesting its greater influence in the clustering process. Interestingly, in *TS k-means (Aggregated)*, the importance was reversed, with the grouped departure week carrying 75% importance compared to the destination country. This shows that the clustering process can be influenced by either variable, depending on the specific clustering approach.

Therefore, it is likely that these variables are interconnected. This can be seen from the above random forecast analysis but also from Figure 5.3, as all the variables of destination countries and grouped departure weeks considered, were almost never exclusively associated with a single cluster. In this case, almost all the

destination countries are part of all the clusters and similarly for the grouped departure weeks. This observation underscores the interplay between these variables, meaning that the relationship between destination countries and grouped departure weeks is not easily separable into distinct groups. For example, simply grouping all booking patterns to Greece or bundling departure weeks together may not always ensure effective grouping of similar booking patterns. This further highlights that a comprehensive and unsupervised approach such as clustering can reveal more complex patterns than a regular ”grouping” of destination countries or departure weeks, where the patterns may not be as cohesive within each group.

Another important aspect to consider for the different methods used to cluster, is the extent to which the booking patterns are being grouped similarly when looking at the cluster labels. In assessing the similarity of clustering assignments across different methods, the Adjusted Rand Index (ARI) was employed. The ARI measures the similarity between two clusterings by comparing pairs of sample labels, adjusted for chance, regardless of the specific numbered cluster labels assigned to each. A higher ARI suggests higher agreement (Santos & Embrechts, 2009).

Table 5.2: Adjusted Rand Index (ARI) Comparisons

Comparisons	**Adjusted Rand Index (ARI)**
TS k-means and Feature Clustering	0.337
TS k-means and TS k-means (Aggregated)	0.213
TS k-means (Aggregated) and Feature Clustering	0.076

Interestingly, the ARI results indicate that TS k-means and Feature Clustering have a higher degree of agreement, signifying a stronger alignment between these clustering approaches. This was unexpected, because they employ two very different methods. On the other hand, the comparison of TS k-means (Aggregated) and TS k-means shows moderate agreement and finally TS k-means (Aggregated) and Feature Clustering have the lowest level of agreement. In contrast, it was expected that TS k-means and TS k-means (Aggregated), would be more similar since they employ the same algorithm (TS k-means). Important to consider is that none of the combination of clustering methods showcased significant similarity. This divergence of clustering labels could be advantageous, as it leads to distinct groupings and therefore distinct models trained for each cluster. Therefore this can offer valuable insights into which clustering method would yield optimal results and what is the optimal grouping for these booking patterns.

5.2 Model Results

During the hyperparameter selection process for each model, a five-fold cross-validation Gridsearch was employed in the training dataset to minimize the custom scorer built, namely minimizing the MSE and MAE. The chosen models and optimal parameters for this thesis are as follows:

Table 5.3: Chosen models and Parameters

General Model	Chosen Models / Parameters
Multiple Linear Regression	Regression on all the booking patterns; *OLS* outperformed *OLS (high volume)*
Weighted Linear Regression	Weights proportional to the total passengers; *WLS (proportional)* outperformed *WLS (doubled)*
Ridge Regression	lambda : 8
Regression Tree	max_depth : 50, min_samples_leaf: 15, max_features= 0.5 , min_samples_split=5.

Note: *OLS* outperforming *OLS (high volume)* may be attributed to the prior exclusion of holiday packages with fewer than 26 passengers from the training set.

For the ensemble models the optimal parameters used for every cluster as well as the weight selection that was done to minimize the MAE and MSE, can be found in the Appendix C. Evidently for the regression tree ensembles, weights assigned to the cluster forecasts are lower (0.4 or 0.5) compared to those of the linear regression ensembles (0.8 for all cases). This may arise because regression trees require more data to achieve better fitting and clusters typically represent a relatively smaller subset of the dataset. Therefore, it was observed that forecasts from the general model carried more weight in such cases.

Below are two tables comparing the performance of the 11 models on the test dataset. *Table 5.4* presents the Mean Squared Error (MSE) and Weighted Mean Squared Error (WMSE), while the *Table 5.5* shows the Mean Absolute Error (MAE) and Weighted Mean Absolute Error (WMAE). The statistical significance of the errors measured was tested using the Wilcoxon signed-rank test. The Wilcoxon signed-rank test is a non-parametric statistical test, that assesses whether the median of paired differences between two sets of observations is zero (Woolson, 2005). In this case for every error measure used, the Wilcoxon signed-rank test will perform a comparison for all the errors by calculating the absolute differences between two forecasts and will assign ranks based on the magnitude of these differences. The test statistic is defined as the summation of all the ranks and will be compared to the critical value of the significance level given, to assess the statistical difference of differences in forecasting performance across the entire dataset.

For the following tables, the $*$ shows whether there was a statistical difference between the error metrics (MSE and MAE) of that model and the Average Model, using a significance level of 0.05. To determine the better model, the one with the lower error score is preferred if the results are significant.

Table 5.4: Mean Squared Error and Weighted Mean Squared Error of Models

Model	MSE	WMSE
Average Model	0.000910	0.000269
Linear Regression $*$	0.000915	0.000277
Weighted Linear Regression $*$	0.000909	0.000267
Ridge $*$	0.000915	0.000278
Regression Tree $*$	0.000918	0.000279
Linear Regression & Feature Clustering	0.000905	0.000267
Linear Regression & TS k-means (Aggregated) $*$	0.000853	0.000238
Linear Regression & TS k-means $*$	0.000895	0.000249
Regression Tree & Feature Clustering $*$	0.000908	0.000273
Regression Tree & TS k-means (Aggregated) $*$	0.000870	0.000250
Regression Tree & TS k-means $*$	0.000903	0.000261

Regarding the MSE; It can be seen that regarding the general forecasting models, only the weighted linear regression managed to slightly outperform the benchmark average model, given the significance level. It is also evident that the weighted linear regression model outperformed the simple linear regression model, showing that taking into account the number of passengers of each holiday package using weights can yield better results. Ridge and linear regression performed the same, suggesting that regularization may not be necessary in this case.

Regarding the ensemble models built and the forecasting errors, all the ensembles managed to outperform the benchmark model, most of them significantly. The linear regression based ensembles outperform the regression tree ones, when compared pair-wise; meaning if the linear regression & TS k-means is compared with the regression tree & TS k-means. In terms of the MSE, for both cases of linear and nonlinear based ensembles, the TS k-means (Aggregated) clustering was the best method used.

Looking at the WMSE, it can be seen that the errors are much smaller. This occurs because this metric scales each error by a fraction, which corresponds to the number of passengers in each holiday package divided by the total number of passengers. The insights generated are similar to the ones for the MSE, namely that most of the general forecasting models are not able to outperform the benchmark model, except weighted linear regression, but not significantly. For the WMSE, most of the ensembles manage to outperform the benchmark model, with the linear regression-based ones showing lower errors than the regression tree ones with the best clustering method being again TS k-means (Aggregated).

For the ranking of the cluster ensembles, both for the MSE and WMSE, the best method for both the linear regression and regression tree based ensembles was the TS k-means(Aggregated) one followed by TS k-means and lastly the Feature Clustering.

To assess whether the inclusion of the forecast from each cluster can yield better results than the general models, it is observed that the Linear Regression model was outperformed by the all three ensemble models that incorporated clustering and linear regression for both the WMSE and MSE. For the regression trees, the

same holds; These findings, highlight the effect of using clustering to capture different patterns in the data at a cluster-level, generally leading to more accurate forecasting.

Important to note as well, is that there were not that many differences regarding the ranking of the models for the MSE and WMSE.

Table 5.5: Mean Absolute Error and Weighted Mean Absolute Error of Test Set

Model	MAE	WMAE
Average Model	0.009661	0.006442
Linear Regression $*$	0.009891	0.006677
Weighted Linear Regression $*$	0.009847	0.006575
Ridge $*$	0.009893	0.006680
Regression Tree $*$	0.009828	0.006673
Linear Regression & Feature Clustering $*$	0.009736	0.006610
Linear Regression & TS k-means (Aggregated) $*$	0.009385	0.006249
Linear Regression & TS k-means	0.009666	0.006381
Regression Tree & Feature Clustering $*$	0.009826	0.006647
Regression Tree & TS k-means (Aggregated) $*$	0.009511	0.006357
Regression Tree & TS k-means $*$	0.009723	0.006461

Regarding the MAE; it can be seen that the average model performs particularly well; only two of the ten models managed to outperform it significantly. The two models are the ensembles that utilize TS k-means (Aggregated), both the linear regression and regression tree. For the other general models, the best performing model is the regression tree. In this case as well, the linear regression based ensembles outperform the regression tree ones when compared pairwise and the ranking of the clustering methods is TS k-means (Aggregated), followed by TS k-means and then the Feature Clustering.

Regarding the WMAE; the only models that manage to outperform the benchmark model are three ensembles that utilize TS k-means (Aggregated) and TS k-means. Regarding the other general models; the best performing one is weighted linear regression followed by regression tree (as opposed to the MAE), illustrating that the weighted linear regression performs better at predicting booking patterns of higher total passengers. Again, for the ensemble models the linear regression based ones outperform regression tree ones.

Below is the overall ranking of each model, determined by calculating the percentage deviation of each model from the benchmark model. This approach accounts for the different scales of MSE and MAE by summing their respective percentage deviations from the benchmark performance. As can be seen, five out of the six ensemble models (green color) managed to outperform or match the benchmark average model (blue color). All the other general models (gray color) are ranked below the benchmark model. The best performing models are evidently the ensemble models that utilized the TS k-means algorithm, the aggregated clustering better than the non-aggregated one. Also the linear regression based ensembles outperform the regression trees ones when utilizing the same clustering method. These results show that clustering is an efficient method to capture trends in the booking patterns and that these patterns can be well captured by a linear based model.

Figure 5.4: Overal ranking of models

Below, Figure 5.5 shows an example graph of the forecasted booking patterns of the ensemble models that utilize clustering and the actual pattern (depicted with a black color). It can be observed that the forecasted booking patterns manage to adequately capture the variation and peaks in the observed values.

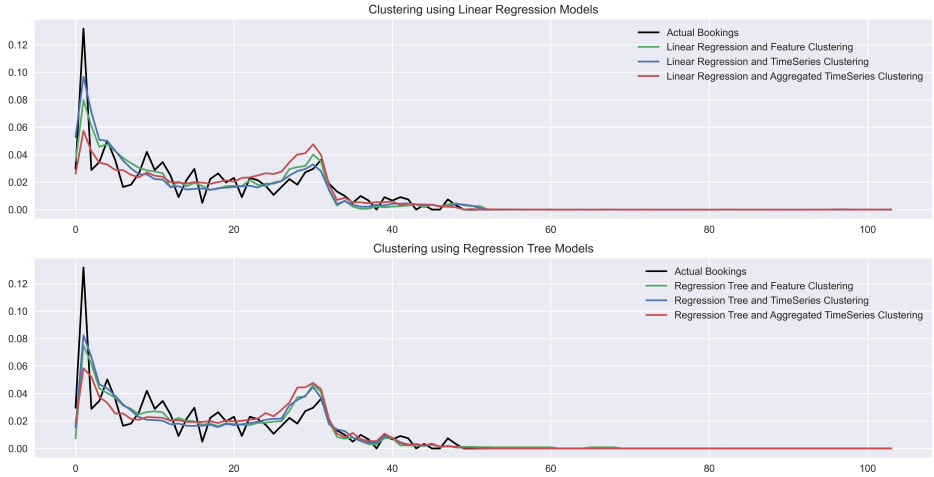


Figure 5.5: Forecasted Booking Patterns

Project deliverable

All the resources needed to reproduce the results of this thesis can be found in this **GitHub repository** and can be used by Sunweb for future research. This repository includes all the trained models, analyses, figures, and results of the thesis organized in different sections for easy understanding and implementation.

6. Discussion

As stated, the aim of this thesis is to assess the performance of forecasting models in comparison to ensembles that are also trained on clustered data. The primary objective is to determine whether training models at the cluster level, where more similar booking patterns are expected, allows the models to better capture the underlying trends. Multiple different forecasting models and clustering methods were utilized in an attempt to conclude on an optimal method that minimizes forecasting errors.

Firstly, the different clustering methods, generally produced different clustering labels for the booking patterns, as can be seen by the Table 5.2: ARI Comparisons. Within each clustering methodology, the medoid curves, which are a good tool to understand the patterns associated with every cluster, in general showed a different booking behavior; in terms of when the bookings start to accumulate, the rate at which they accumulate every week, if there are significant jumps and the trends regarding last minute reservations. This is in line with Schwartz (2008), who argued that distinct booking patterns exist.

In this analysis, it was also attempted to understand the distribution of characteristics of the holiday packages, such as their destination country and departure week in terms of the clusters. This was not a straightforward task, meaning that in general all the holiday packages' characteristics, were associated mostly with all the cluster labels in each of the cluster methods. This actually was interesting, because it signifies that clustering; being an unsupervised method, can yield better groupings based on the booking patterns than a simple grouping method. So for example, grouping all the holiday packages per destination country will not guarantee that the booking patterns in that group will be more cohesive with each other.

Another interesting insight regarded the departure weeks. In general, it was seen that different holiday packages can be well summarized by their week of departure, since the "naive" benchmark model per departure week, managed to perform relatively well for all the error metrics used. This means that, holiday packages departing for example in the 33rd week of the year, regardless of the destination exhibit a strong forecasting pattern. This was also why in the modeling phase of the general models; no dummy variables were introduced for the destination country, and it was solely based on the booking week and weeks in advance. When trying to delve deeper and understand how the booking patterns of departure weeks differ, it was seen in Figure 5.2: Dendogram of Departure Weeks, that certain sequential and non-sequential weeks were similar to each other; For example, holiday packages whose departure was in the 2nd week, exhibited similar booking behavior as the ones departing in departure week 17. Definitely, more domain knowledge is needed to understand why this occurs, but these insights can be valuable for Sunweb.

Regarding the model performance using Figure 5.4: Overal Ranking of models; it was shown that the ensemble models that were also trained on clustered data have outperformed the other general forecasting models. Specifically, five out of the six ensemble models were ranked as the top performers and they all managed to outperform or match the benchmark average model. In particular the ensembles utilizing the TS k-means algorithm (aggregated and non) performed the best. While this thesis focused on building ensemble

models for the forecasts, rather than just forecasting per cluster as other studies in the field (Viverit et al., 2023; Bandara et al., 2020), this methodology still produced the same results. Additionally, when assessing the models, both linear regression and regression trees with their respective ensembles across different cluster methods, the ensembles showed improved results. These results show that training models at a cluster level and incorporating this in the final forecast can have significant positive impact in forecasting.

Comparing the performance of the linear and non-linear based ensembles; it is concluded for this thesis, that the linear based ensembles performed better than the non-linear ones, when the same clustering was used. As can be seen from the Figure 5.5, the regression trees sometimes, were able to better capture peaks (for example in this case the last minute peak 1 week before departure), that the linear ensembles did not. This is in line with results from Masini et al. (2021) that observed that these models were able to capture more complex trends than the the linear ones. However, in this thesis the linear models performed better, possibly due to the strength of the booking patterns (not so many unexpected patterns in the bookings).

Furthermore, the general models did not manage to outperform the benchmark average model. For the general models, the best performing model was the weighted linear regression followed by regression trees and linear regression. The weighted linear regression outperformed the linear regression, indicating that assigning weights to the the booking patterns based on the total passengers helps the models to detect trends. Apparently, the more passengers, the more consistent the booking patterns, and therefore, training should prioritize high-volume holiday packages.

Summary of Results

- Clustering is a valuable method in booking pattern forecasting.
- Ensemble models that utilized clustering outperformed all other models, including the benchmark model, which was not always outperformed by the general models.
- Clustering booking patterns is effective for grouping similar booking behaviors due to its unsupervised nature.
- The best performing clustering method was the TS k-means algorithm.
- Linear regression-based models generally outperformed non-linear ones.

Limitations and Future Research

A limitation in this thesis, was the limited understanding of the grouped departure week and the destination country in each of the clustering assignments. The three clustering methods used, did not produce easily interpretable results; for example concluding that the majority of the holiday packages to a certain destination or departure week are part of one cluster. However, this limitation can also be viewed as a strength, as clustering being an unsupervised method; can potentially reveal unexpected groupings that can yield better results.

Another limitation of this thesis was its assumption that all holiday packages were of equal value and represented identical holiday experiences for the consumer. An addition to that could be, following Kaya et al. (2022), to include feature vector characteristics for each holiday package such as length of stay, a type of vacation index (e.g. family-friendly), hotel facilities etc. to group the holiday packages in a different way and understand their booking patterns.

An interesting area to explore would be combining forecasts of non-linear and linear models for the general and cluster forecast; for example using a non-linear model for the general forecast and a linear model for the cluster forecast and vice versa. Linear models proved to be valuable for capturing simple and linear relationships between variables, but regression trees also managed to capture some non-linear complexities in the booking patterns. Therefore combining the two approaches using ensemble modeling, might yield better results. Additionally, for the ensemble modeling, other methods such as training a Voting Regressor, can potentially yield lower forecasting errors that the weight optimization that has been done in this thesis.

Regarding the clustering; Future research can also consider including other curve characteristics extracted from the time series for Feature Clustering, such as Entropy and Non-Linearity. Also, including new curve characteristics specific to booking patterns, based on domain knowledge, such as the frequency of reaching certain significant peaks, can also be beneficial. Additionally, for TS k-means (Aggregated), which was the best performing clustering, more options can be explored regarding the groupings of booking weeks. Rather than uniformly grouping them into intervals of two sequential booking weeks, an alternative approach could involve grouping fewer booking weeks together as the departure date approaches, while more weeks could be grouped together further from the departure date. This perhaps can help cluster the curves by detecting other patterns that are important for Sunweb, such as last-minute bookings.

An area of future research following this thesis, can be to integrate in the analysis another dataset of the average weekly pricing for each holiday package, even when no bookings were made in a booking week. This way, price sensitivity and other dynamics can be explored to unveil other potential trends. In an industry where pricing fluctuates daily, such additional data could prove to have high explanatory power. Another area that can be explored is to include external data. The overall effects of popularity and demand of a destination based on Search Engine data such as Google Trends has shown to be very effective (Pan et al., 2012).

Statement of Work

All sections in this thesis are my individual contribution. I, Anni Tziakouri, take full responsibility for all work presented in this thesis.

Statement of Originality

This document is written by Anni Tziakouri, who declares to take full responsibility for the contents of this document. I declare that the text and the work presented in this document are original and that no sources other than those mentioned in the text and its references have been used in creating it. I have not used generative AI (such as ChatGPT) to generate or rewrite text. UvA Economics and Business is responsible solely for the supervision of completion of the work and submission, not for the contents.

7 REFERENCES
References

Aghabozorgi, S., Shirkhorshidi, A. S., & Wah, T. Y. (2015). Time-series clustering–a decade review. *Information systems*, *53*, 16–38.

Ahmad, W. K. A. W., & Ahmad, S. (2013). Arima model and exponential smoothing method: A comparison. *AIP conference proceedings*. https://doi.org/10.1063/1.4801282

Ahmed, N. K., Atiya, A. F., Gayar, N. E., & El-Shishiny, H. (2010). An empirical comparison of machine learning models for time series forecasting. *Econometric reviews*, *29*(5-6), 594–621. https://doi.org/10.1080/07474938.2010.481556

Amigó, E., Gonzalo, J., Artiles, J., & Verdejo, F. (2009). A comparison of extrinsic clustering evaluation metrics based on formal constraints. *Information retrieval*, *12*, 461–486.

Bandara, K., Bergmeir, C., & Smyl, S. (2020). Forecasting across time series databases using recurrent neural networks on groups of similar series: A clustering approach. *Expert systems with applications*, *140*, 112896. https://doi.org/10.1016/j.eswa.2019.112896

Banerjee, N., Morton, A., & Akartunalı, K. (2020). Passenger demand forecasting in scheduled transportation. *European journal of operational research*, *286*(3), 797–810. https://doi.org/10.1016/j.ejor.2019.10.032

Ben-Akiva, M., et al. (1987). Improving airline passenger forecasts using reservation data. *Presentation at Fall ORSA/TIMS Conference, St. Louis, MO*.

Bismi, I. (2023). Linkage Methods, single linkage, clustering, hierarchical clustering — Medium. https://medium.com/@iqra.bismi/different-linkage-methods-used-in-hierarchical-clustering-627bde3787e8

Cerqueira, V., Torgo, L., & Soares, C. (2019). Machine learning vs statistical methods for time series forecasting: Size matters. *arXiv preprint arXiv:1909.13316*.

De Sá, J. R. (1987, February). Reservations forecasting in airline yield management. https://dspace.mit.edu/handle/1721.1/68058

Ezugwu, A. E., Ikotun, A. M., Oyelade, O. O., Abualigah, L., Agushaka, J. O., Eke, C. I., & Akinyelu, A. A. (2022). A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects. *Engineering applications of artificial intelligence*, *110*, 104743. https://doi.org/10.1016/j.engappai.2022.104743

Faraway, J., & Chatfield, C. (1998). Time series forecasting with neural networks: A comparative study using the air line data. *Journal of the Royal Statistical Society Series C: Applied Statistics*, *47*(2), 231–250.

Fildes, R., & Lusk, E. J. (1984). The choice of a forecasting model. *Omega*, *12*(5), 427–435. https://doi.org/10.1016/0305-0483(84)90042-2

Fulcher, B. D., & Jones, N. S. (2014). Highly comparative feature-based time-series classification. *IEEE Transactions on Knowledge and Data Engineering*, *26*(12), 3026–3037.

Geurts, M., Box, G. E. P., & Jenkins, G. M. (1977). Time Series Analysis: Forecasting and control. *Journal of marketing research*, *14*(2), 269. https://doi.org/10.2307/3150485

Goia, A., May, C., & Fusai, G. (2010). Functional clustering and linear regression for peak load forecasting. *International journal of forecasting*, *26*(4), 700–711. https://doi.org/10.1016/j.ijforecast.2009.05.015

Gordon, A. D., Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). Classification and regression trees. *Biometrics*, *40*(3), 874. https://doi.org/10.2307/2530946

Huang, L., & Zheng, W. (2021). Novel deep learning approach for forecasting daily hotel demand with agglomeration effect. *International journal of hospitality management*, *98*, 103038. https://doi.org/10.1016/j.ijhm.2021.103038

Huang, X., Ye, Y., Xiong, L., Lau, R. Y. K., Jiang, N., & Wang, S. (2016). Time series k -means: A new k -means type smooth subspace clustering for time series data. *Information sciences*, *367-368*, 1–13. https://doi.org/10.1016/j.ins.2016.05.040

Hyndman, R. J., Kostenko, A. V., et al. (2007). Minimum sample size requirements for seasonal forecasting models. *foresight*, *6*(Spring), 12–15.

Ikotun, A. M., Ezugwu, A. E., Abualigah, L., Abuhaija, B., & Heming, J. (2023). K-means clustering algorithms: A comprehensive review, variants analysis, and advances in the era of big data. *Information sciences*, *622*, 178–210. https://doi.org/10.1016/j.ins.2022.11.139

Jain, A. K. (2010). Data clustering: 50 years beyond K-means. *Pattern recognition letters*, *31*(8), 651–666. https://doi.org/10.1016/j.patrec.2009.09.011

Kaya, K., Yılmaz, Y., Yaslan, Y., Öğüdücü, Ş. G., & Çıngı, F. (2022). Demand forecasting model using hotel clustering findings for hospitality industry. *Information processing management*, *59*(1), 102816. https://doi.org/10.1016/j.ipm.2021.102816

Kodinariya, T. M., Makwana, P. R., et al. (2013). Review on determining number of cluster in k-means clustering. *International Journal*, *1*(6), 90–95.

Kouprioucha, L., Van Der Rest, J.-P., & Schwartz, Z. (2014). On revenue management and the use of occupancy forecasting error measures. *International journal of hospitality management*, *41*, 104–114. https://doi.org/10.1016/j.ijhm.2014.05.002

Lee, A. O. (1990, January). Airline reservations forecasting–probabilistic and statistical models of the booking process. https://dspace.mit.edu/handle/1721.1/68100

Lewis, R. J. (2000). An introduction to classification and regression tree (cart) analysis. *Annual meeting of the society for academic emergency medicine in San Francisco, California*, *14*.

Li, K., Li, W.-X., Liang, C., & Wang, B. (2019). Intelligence in Tourism Management: A Hybrid FOA-BP Method on Daily Tourism Demand Forecasting with Web Search Data. *Mathematics*, *7*(6), 531. https://doi.org/10.3390/math7060531

Liao, T. W. (2005). Clustering of time series data—a survey. *Pattern recognition*, *38*(11), 1857–1874.

Ma, M., Liu, J., & Cao, J. (2014). Short-Term forecasting of railway passenger flow based on clustering of booking curves. *Mathematical problems in engineering*, *2014*, 1–8. https://doi.org/10.1155/2014/707636

Makridakis, S. (1996). Forecasting: its role and value for planning and strategy. *International journal of forecasting*, *12*(4), 513–537. https://doi.org/10.1016/s0169-2070(96)00677-2

Makridakis, S., Andersen, A., Carbone, R. F., Fildes, R., Hibon, M., Lewandowski, R., Newton, J., Parzen, E., & Winkler, R. L. (1982). The accuracy of extrapolation (time series) methods: Results of a forecasting competition. *Journal of forecasting*, *1*(2), 111–153. https://doi.org/10.1002/for.3980010202

Makridakis, S., Spiliotis, E., & Assimakopoulos, V. (2020). The M4 Competition: 100,000 time series and 61 forecasting methods. *International journal of forecasting*, *36*(1), 54–74. https://doi.org/10.1016/j.ijforecast.2019.04.014

Masini, R., Medeiros, M. C., & Mendes, E. (2021). Machine learning advances for time series forecasting. *Journal of economic surveys*, *37*(1), 76–111. https://doi.org/10.1111/joes.12429

Müller, M. (2007). *Information retrieval for music and motion* (Vol. 2). Springer.

Nieto, M. R., & Carmona-Benítez, R. B. (2018). ARIMA + GARCH + Bootstrap forecasting method applied to the airline industry. *Journal of air transport management*, *71*, 1–8. https://doi.org/10.1016/j.jairtraman.2018.05.007

Pan, B., Wu, D. C., & Song, H. (2012). Forecasting hotel room demand using search engine data. *Journal of hospitality and tourism technology*, *3*(3), 196–210. https://doi.org/10.1108/17579881211264486

Pereira, L. N., & Cerqueira, V. (2021). Forecasting hotel demand for revenue management using machine learning regression methods. *Current issues in tourism*, *25*(17), 2733–2750. https://doi.org/10.1080/13683500.2021.1999397

Schwartz, Z. (2006). Advanced booking and revenue management: Room rates and the consumers' strategic zones. *International journal of hospitality management*, *25*(3), 447–462. https://doi.org/10.1016/j.ijhm.2005.02.002

Schwartz, Z. (2008). Time, price, and advanced booking of hotel rooms. *International Journal of Hospitality Tourism Administration*, *9*(2), 128–146. https://doi.org/10.1080/15256480801907885

Schwartz, Z., & Hiemstra, S. J. (1997). Improving the accuracy of hotel Reservations Forecasting: Curves Similarity approach. *Journal of travel research*, *36*(1), 3–14. https://doi.org/10.1177/004728759703600102

Sunweb Group. (2023). Sustainability report. https://prodsunwebgroupstore.blob.core.windows.net/media/2024/05/SunwebGroup_Sustainability-Report-2023.pdf

Taylor, S. J., & Letham, B. (2018). Forecasting at scale. *The American statistician*, *72*(1), 37–45. https://doi.org/10.1080/00031305.2017.1380080

Thompson, M. (1982). Regression methods in the comparison of accuracy. *Analyst (London. 1877. Online)/Analyst*, *107*(1279), 1169. https://doi.org/10.1039/an9820701169

Tse, T., & Poon, Y.-T. (2015). Analyzing the use of an advance booking curve in forecasting hotel reservations. *Journal of travel tourism marketing*, *32*(7), 852–869. https://doi.org/10.1080/10548408.2015.1063826

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

Wang, X., Smith, K., & Hyndman, R. (2006). Characteristic-Based clustering for time series data. *Data mining and knowledge discovery*, *13*(3), 335–364. https://doi.org/10.1007/s10618-005-0039-x

Weatherford, L., & Kimes, S. E. (2003). A comparison of forecasting methods for hotel revenue management. *International journal of forecasting*, *19*(3), 401–415. https://doi.org/10.1016/s0169-2070(02)00011-0

Weatherford, L. R., Gentry, T. W., & Wilamowski, B. (2003). Neural network forecasting for airlines: A comparative analysis. *Journal of Revenue and Pricing Management*, *1*, 319–331.

Webb, T., Schwartz, Z., Xiang, Z., & Singal, M. (2020). Revenue management forecasting: The resiliency of advanced booking methods given dynamic booking windows. *International journal of hospitality management*, *89*, 102590. https://doi.org/10.1016/j.ijhm.2020.102590

Weisberg, S. (2005). *Applied linear regression* (Vol. 528). John Wiley & Sons.

Wickham, R. R. (1995, January). Evaluation of forecasting techniques for short-term demand of air transportation. https://dspace.mit.edu/handle/1721.1/68101

Woolson, R. F. (2005). Wilcoxon signed-rank test. *Encyclopedia of Biostatistics*, *8*.

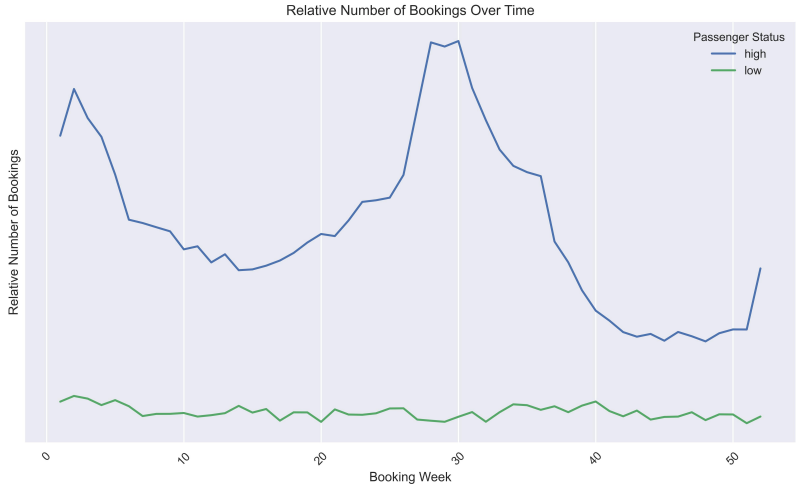
Wu, H., & Levinson, D. (2021). The ensemble approach to forecasting: A review and synthesis. *Transportation Research Part C: Emerging Technologies*, *132*, 103357.

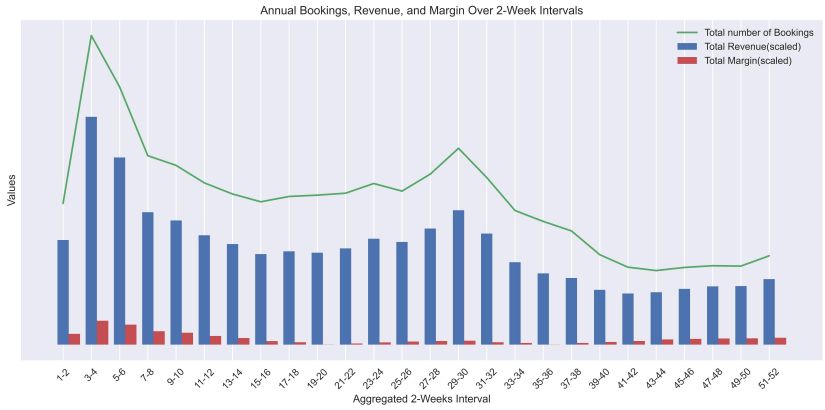
Zhang, Y. (2019). *Forecasting hotel demand using machine learning approaches* [Doctoral dissertation, Cornell University].

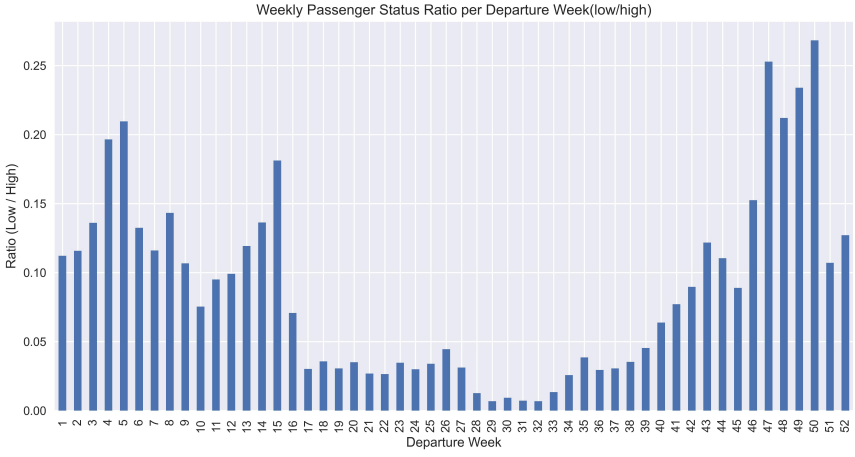
Zuccarelli, E. ”. (2021). Performance Metrics in ML - Part 3: Clustering — towards Data Science. https://towardsdatascience.com/performance-metrics-in-machine-learning-part-3-clustering-d69550662dc6

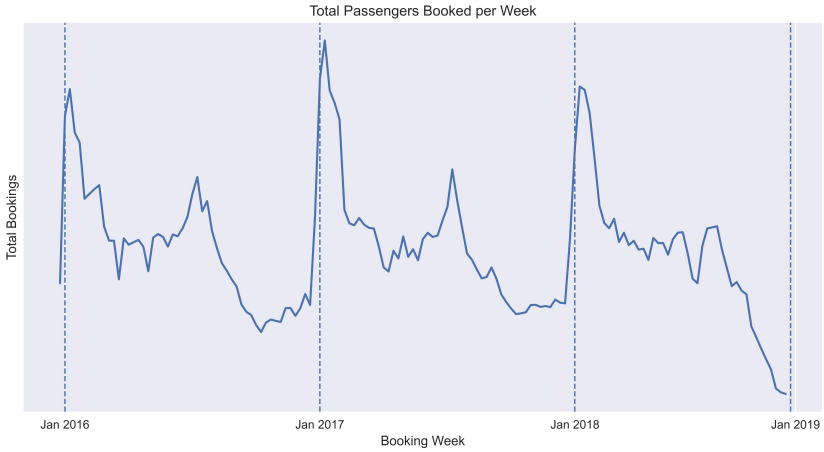
A. Data Visualizations

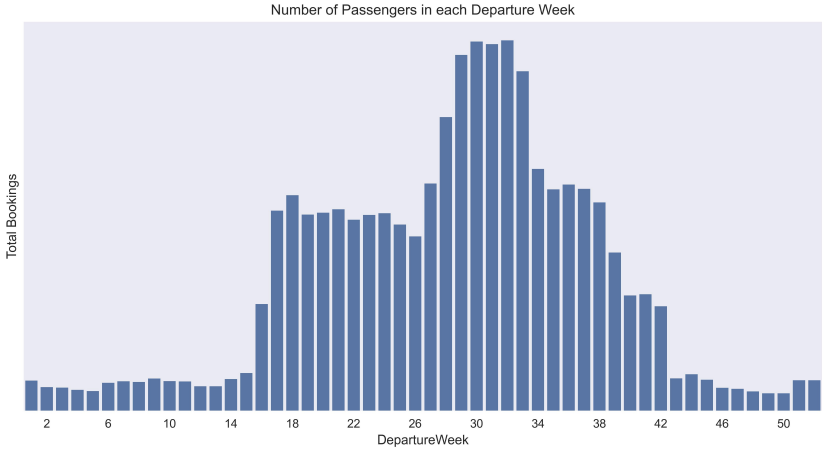
Note: The number of total passengers has been hidden due to confidentiality.

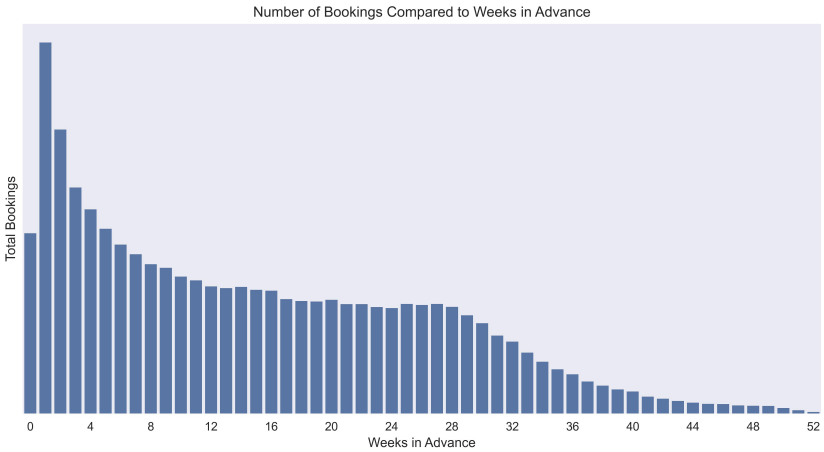




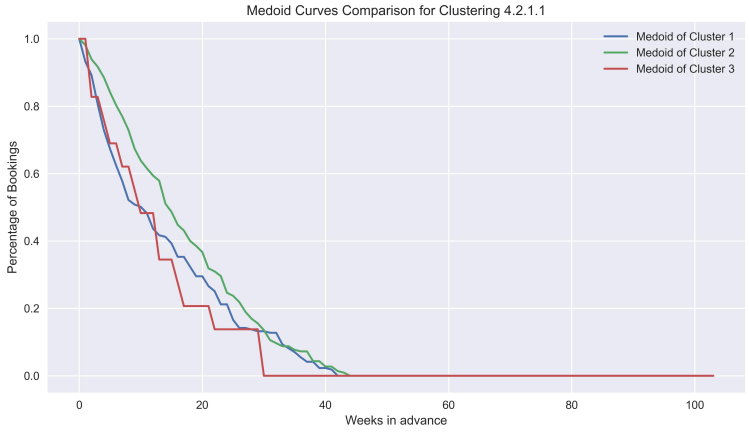


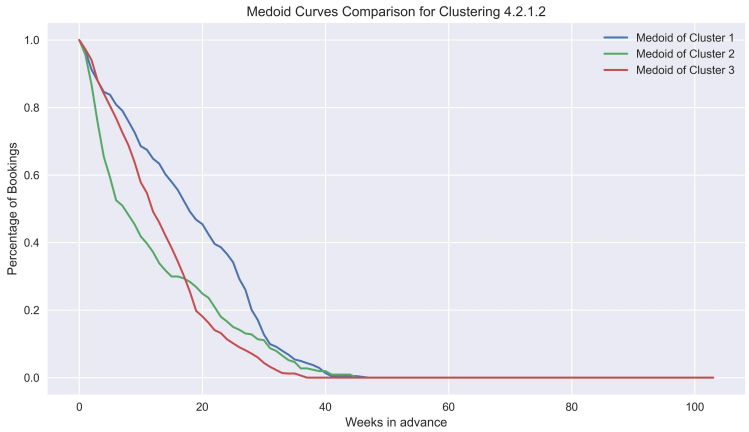

Total Passengers Booked per Week
Total Bookings
Jan 2016
Jan 2017
Jan 2018
Jan 2019
Booking Week

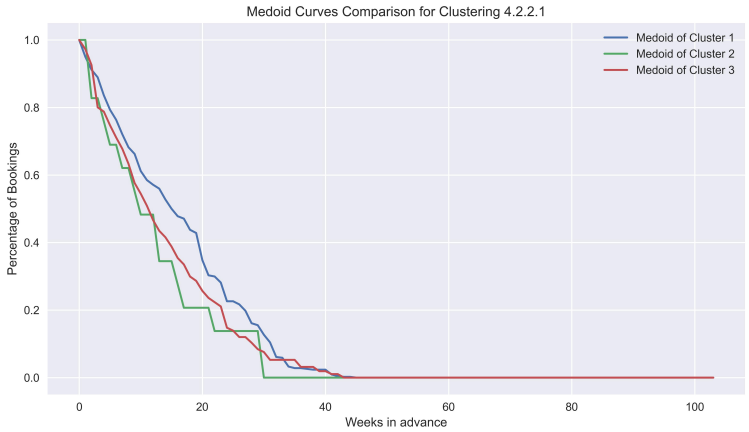

Number of Passengers in each Departure Week
Total Bookings
2
6
10
14
18
22
26
30
34
38
42
46
50
DepartureWeek


Number of Bookings Compared to Weeks in Advance
Total Bookings
0
4
8
12
16
20
24
28
32
36
40
44
48
52
Weeks in Advance

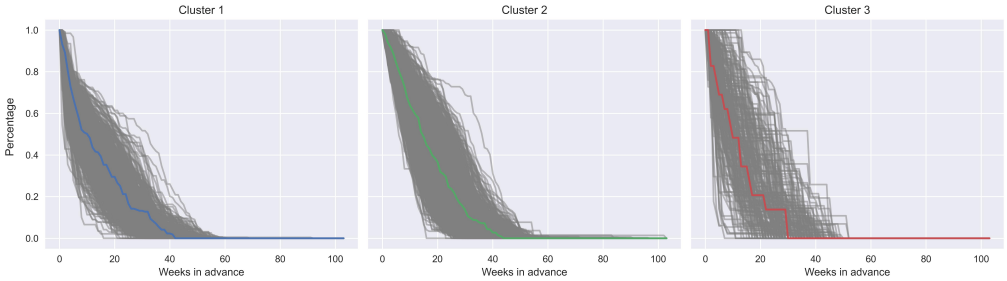
B. Clustering Results



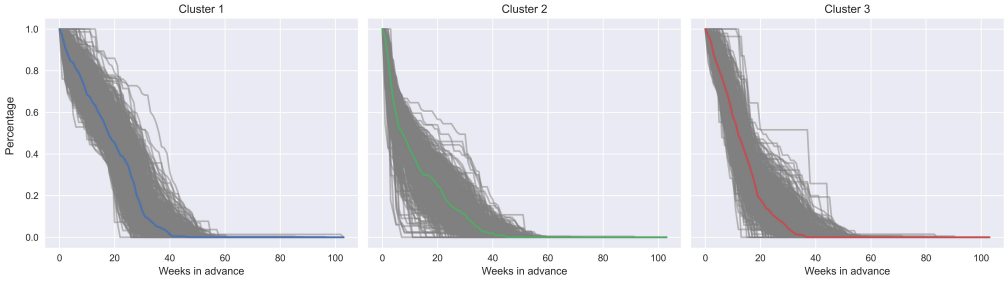




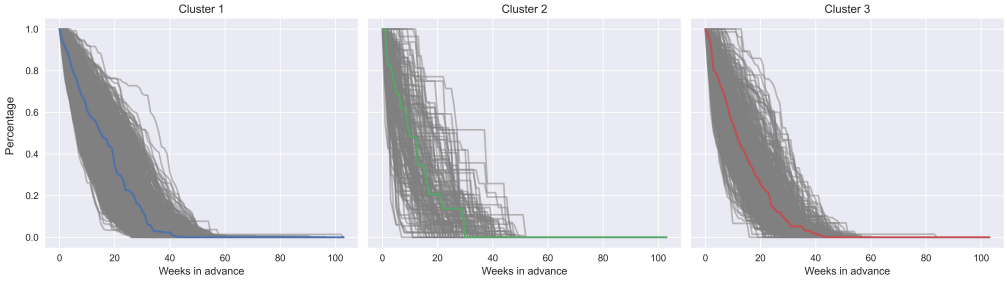
Clustering 4.2.1.1



Clustering 4.2.1.2



Clustering 4.2.2.1



C. Ensemble models

Table C.1: Ensemble Model Weights

Model	Weight General Forecast	Weight Cluster Forecast
Linear Regression & Feature Clustering	0.2	0.8
Linear Regression & TS k-means (Aggregated)	0.2	0.8
Linear Regression & TS k-means	0.2	0.8
Regression Tree & Feature Clustering	0.5	0.5
Regression Tree & TS k-means (Aggregated)	0.4	0.6
Regression Tree & TS k-means	0.5	0.5

Table C.2: Clustering and Parameters Used

Clustering	Parameters used
TS k-means	
Cluster 1:	max_depth=30, max_features=0.5, min_samples_leaf=20 , min_samples_split =5
Cluster 2:	max_depth=40, max_features=0.5, min_samples_leaf=10, min_samples_split=5
Cluster 3:	max_depth=45, max_features=0.5, min_samples_leaf=20, min_samples_split=5
TS k-means (Aggr)	
Cluster 1:	max_depth=30, max_features=0.5, min_samples_leaf=10, min_samples_split=5
Cluster 2:	max_depth=35, max_features=0.5, min_samples_leaf=15, min_samples_split=5
Cluster 3:	max_depth=45, max_features=0.5, min_samples_leaf=25, min_samples_split=5
Feature Clustering	
Cluster 1:	max_depth=45, max_features=0.5, min_samples_leaf=10, min_samples_split=5
Cluster 2:	max_depth=45, max_features=0.5, min_samples_leaf=15, min_samples_split=5
Cluster 3:	max_depth=45, max_features=0.5, min_samples_leaf=25, min_samples_split=5