A Work Project, presented as part of the requirements for the Award of a Master's degree in Business Analytics from the Nova School of Business and Economics.

NOS & Nova SBE Project-Based Learning - Predicting the Volume Of Customers to Flag for Call Center's Specialized Team

Enhancing the performance of time series models through Unsupervised Outlier Detection and Treatment Techniques

AUTHOR

Victoria Muguerza

Work project carried out under the supervision of:

Susana Lavado &

Gustavo Pereira

29-01-2025

Abstract

This thesis is part of a broader research effort extending the outcomes of the year-long Project-Based Learning initiative with NOS, a prominent telecommunications company in Portugal, focusing on optimizing the number of clients that should be flagged for specialized call center teams, to increase clients' satisfaction. While the larger thesis addresses both model performance and explainability, this specific work focuses on improving model performance through outlier detection. By refining the handling of outliers, this study contributes to more accurate predictions, ultimately enhancing the effectiveness of client selection.

Keywords: Call Centers, Time Series Forecasting, Prediction Modeling, Unsupervised Outlier Detection, Trimming, Winsorization, Robust Estimation, SARIMA, XGBoost

Acknowledgements

I am deeply grateful to my thesis partners, Hendrik Künnemann, João Miranda, and João Penedo, as well as my family and friends, for their unwavering support and encouragement throughout this journey. I extend my sincere appreciation to my advisors, Susana Lavado and Gustavo Pereira, whose insightful guidance and constructive feedback were invaluable during both the thesis and PBL phases. Finally, I would like to thank Tom Göring, a dedicated and collaborative member of our PBL team, for his significant contributions to the project.

This work used infrastructure and resources funded by Fundação para a Ciência e a Tecnologia (UID/ECO/00124/2013, UID/ECO/00124/2019 and Social Sciences DataLab, Project 22209), POR Lisboa (LISBOA-01-0145-FEDER-007722 and Social Sciences DataLab, Project 22209) and POR Norte (Social Sciences DataLab, Project 22209).

Contents

1 Group Part **4**

1.1 Introduction 4

1.2 Context 4

1.3 Dataset 6

1.4 Literature Review 7

1.4.1 Time Series Forecasting 8

1.4.2 Time Series Forecasting at Call Centers 11

1.4.3 Time Series Forecasting in Limited Size Datasets 11

1.5 Project-Based Learning Project 12

1.5.1 Target Variable and Data Preprocessing 12

1.5.2 Feature Engineering 14

1.5.3 Methodology 14

1.5.3.1 Target Capping 14

1.5.3.2 Introduction of the Models Deployed 15

1.5.3.3 Model Evaluation Criteria 20

1.5.4 Results from PBL 25

1.5.4.1 Deployable Solution 27

1.5.4.2 Build-Onto Solution 28

2 Enhancing the performance of time series models through Unsupervised Outlier Detection and Treatment Techniques **29**

2.1 Literature review 30

2.1.1 Definitions of outliers 31

2.1.2 Unsupervised outliers' detection techniques 31

2.1.3 Treatment of outliers 34

2.1.4 Implications of treating outliers in the target variable and features . . . 35

2.2 Methodology 36

2.2.1 Distribution analysis and normalization of variables 36

2.2.2 Detection and treatment of outliers 37

2.2.3 Hyperparameter tuning and evaluation 38
2.3 Results 39
2.3.1 Distribution analysis 39
2.3.2 Univariate analysis 40
2.3.3 Multivariate analysis 41
2.4 Discussion of the results 43
2.5 Limitations & Future Work 46
2.6 Conclusion 46

A Appendix - Enhancing the performance of time series models through Unsupervised Outlier Detection and Treatment Techniques 59

A.1 Shapiro-Wilk Test Results 59
A.2 XGBoost Pareto Frontier 61

1 Group Part

1.1 Introduction

In today's competitive telecommunications industry, customer satisfaction is essential for retaining a loyal client base. To maintain strong connections with customers, organizations increasingly rely on call centers (Purba and Hidayati 2020). A well-managed call center can greatly enhance an organization's overall performance by optimizing key metrics such as "service level, cost per contact, customer satisfaction, average handle time, first call resolution, abandon rate, average waiting time, and occupancy rate" (Płaza and Pawlik 2021).

NOS, a leading telecommunications operator in Portugal, was established in 2014 through the merger of ZON and Optimus. Recognizing the strategic importance of call center operations, NOS has made significant investments in data science and AI, among them projects to optimize their call center operations. One such initiative was the Project-Based Learning (PBL) in collaboration with NOVA SBE, where a predictive modeling framework was developed to improve key call center metrics. This project provided an opportunity to explore forecasting techniques and operational improvements, focusing on balancing predictive accuracy and stability. The PBL project served as the foundation for this thesis, providing a starting point for predictive modeling and insights into effective call center management strategies.

This thesis is part of a larger group project and focuses only on section 2. While the broader thesis is organized into four main parts, each addressing different aspects of time series forecasting and explainability, this work covers "Enhancing the Performance of Time Series Models through Unsupervised Outlier Detection and Treatment Techniques", which explores methods for detecting and handling outliers in datasets.

1.2 Context

NOS' call center is divided into two distinct lines: the general and the technical line. The general line handles a wide range of customer inquiries, including billing questions, contract changes and other non-technical issues. The technical line, on the other hand, specializes in

resolving technical issues, such as network issues, service interruptions or other technical difficulties. One of the key challenges within their technical line lies in managing the volume of clients who repeatedly contact the call center to address ongoing issues. These recurrent callers, here referred to as ”relapsing callers”, not only burden the call center's operations but also pose a significant risk of client dissatisfaction and potential churn. To address this pressing issue, NOS has established a specialized unit, the Detractor Squad, dedicated to handling calls from clients with a history of unresolved issues on the technical line. This unit is specifically designed to manage both relapsing clients and first-time callers who present particularly complex problems. By providing tailored solutions, the Detractor Squad aims to resolve these challenging cases and prevent further relapsing.

Managing the workload of the Detractor Squad effectively is critical. By resolving customer issues and preventing future calls customer satisfaction and retention are enhanced, supporting NOS' broader goal of maintaining a competitive edge in the telecommunications sector. Beyond that, there are also financial benefits, as the Detractor Squad contributes to significant cost savings.

A significant challenge in this process is that relapsing clients do not all experience their issues on the same day, making it difficult to forecast the daily workload for the Detractor Squad. The optimal workload for the squad corresponds to a specific number of calls handled per day, referred to as the operational target of daily calls. Achieving this target requires balancing the risks of overestimating or underestimating the workload, which will be addressed later.

To address this difficulty of balancing workload and determining when clients should be redirected to the Detractor Squad, NOS implemented a two-stage approach. The first stage involved the use of a classification model to assign each client a daily score between 0 and 1, reflecting the likelihood of them calling within the next 15 days. These scores were compiled into a list, with clients being prioritized based on their score. Once the likelihood scores were calculated, a time series model was used to forecast the optimal number of clients to flag each day from the ordered list. A flagged client is one identified as having a high probability of making a call, allowing their calls to be automatically prioritized and redirected to the Detractor Squad for ex-

pedited resolution. Since flagged clients might not place their calls on the exact day predicted, the optimal number of clients to flag each day was not fixed and did not directly correspond to the operational target for daily calls. Such variability had to be accounted for by the model. Each forecast was conducted two days in advance to allow for potential operational adjustments based on the prediction.

By combining the outputs of the classification and the time series forecasting model, NOS seeked to direct the appropriate number of calls to the Detractor Squad, ensuring more effective intervention, resource allocation, and ultimately, increased operational efficiency. Accurately estimating the volume of clients to redirect was crucial to ensure the optimal utilization of the Detractor Squad. Both overestimation and underestimation could lead to inefficiencies. Overestimating the number of clients to flag can lead to redirecting calls that do not need the Detractor Squad's attention. These calls occupy slots that could otherwise be allocated to calls that truly require intervention. An underestimation, on the other hand, results in the Detractor Squad having insufficient calls to handle, leading to periods of inactivity. These idle moments are a financial burden, as the team incurs costs without adding value. Since the cost of idle time in the Detractor Squad is higher than the potential cost of clients needing to call again because they were not properly advised initially, overestimation is preferred.

1.3 Dataset

The work presented in this thesis, comprising the group section that describes the work done within the Business Analytics Masters' Project-Based Learning program, which will be further introduced in section 1.5 (Project-Based Learning Project), and the four individual parts rely on historical, pseudo-anonymized data from NOS' call center, covering the period from May 31, 2023, to September 29, 2023. Throughout this timeframe, over 2 million calls were treated by NOS' call center with more than 1.5 million clients.

Almost 20% of all the clients were relapsing at least once during the period, highlighting the business challenge at hand. From a model performance perspective, the existing predictive model did not fully meet its objectives, directing fewer than the target established for the De-

tractor Squad. This discrepancy indicates that, on average, the Detractor Squad has the capacity to handle a greater volume of calls, highlighting the need for model optimization.

One key variable in this dataset for this task is the aforementioned score for each client, reflecting the likelihood of a client calling within the next 15 days and generated by NOS' existing classification model. Additionally, information on whether the client made a call on the preceding day is included, which is crucial for defining the target variable, as explained further in section 1.5.1 (Target Variable and Data Preprocessing). Furthermore, this data captures various aspects of client interactions and service usage, along with past call center statistics, which play an important role in the multivariate models used in this project as described in section 1.5.3.2.4 (Multivariate Models) and in feature engineering discussed in section 1.5.2 (Feature Engineering).

In terms of client interactions, the dataset provides detailed records of calls answered by the call center, including attributes such as the teams handling the calls, call duration, hold times, and the number of transfers involved. Regarding service usage it contains information about the outcomes of house visits by service professionals, including metrics such as wait times for service and the occurrence of the visits. The past call center statistics are organized across multiple time intervals, ranging from a few days to multiple years. These statistics encompass a variety of metrics, such as the number of calls made during each time frame, the total time spent on calls, total hold time, and a heuristic score that assigns greater weight to more recent calls.

1.4 Literature Review

This section provides a comprehensive overview of the methods, models, and advancements in time series forecasting relevant to this study. Starting with a general exploration of time series forecasting, it examines the foundational theories, statistical models, and machine learning techniques that have shaped this field. Specific attention is then given to the application of time series forecasting in call centers, highlighting the models used to predict call volumes and optimize operations. Finally, the section discusses the challenges of forecasting with small datasets

and examines why neural networks may not be suitable in such contexts, setting the foundation for the methodological choices made in this study.

1.4.1 Time Series Forecasting

To effectively manage the dynamic nature of relapsing callers and accurately predict daily call volumes, time series forecasting becomes an essential tool. By analyzing historical data patterns, these methods can assist in anticipating future call volumes (Bastianin, Galeotti, and Manera 2019), thereby optimizing call center operations within the Detractor Squad.

A time series is a sequence of data points recorded at successive points in time (Chatfield 2000). The observations are collected over intervals of time scales such as seconds, minutes, hours, days, months, or years. Each observation y_t represents a value measured at a specific time t and the series $(y_1, y_2, \ldots, y_{t_{\max}})$ captures how these values evolve over time (Hamilton 1994).

The task of time series forecasting is then defined as the attempt to predict data points in this series that are not yet observed, y_s, with $s > t_{\max}$, by leveraging statistical properties present in the historical data up to time $t_{\max}$. These methods rely on historical values of the series, also known as lagged values, and may incorporate external factors or explanatory variables to improve accuracy (Chatfield 2000). The ultimate goal is to analyze the underlying structure of the data, such as trends, seasonality, cyclicity, and irregular fluctuations, to generate meaningful forecasts (Koopman and Lee 2009).

Trends represent long-term directional movements in the data, such as persistent increases or decreases over time (Wu and Zhao 2007). Seasonality captures recurring patterns at fixed intervals, such as daily, monthly, or yearly variations, while cyclicity reflects longer-term oscillations possibly driven by factors like economic or environmental cycles (Davey and Flores 1993). Irregular fluctuations, on the other hand, account for random, unpredictable variations that cannot be explained by these systematic components (Zhang, Ng, and Na 2020). These principles make time series forecasting applicable in fields like finance (e.g., stock prices), economics (e.g., GDP growth rates), and meteorology (e.g., temperature records), where the primary goal is to uncover and understand the underlying structure of the data (Yildiz and Kundakcioglu 2024).

Time series forecasting is a well-established field with a long history of development. Numerous methods and algorithms have been developed to address the task of time series forecasting. However, no single model can consistently provide the best forecasting results for all types of time series data due to the inherent complexity of the data. Additionally, the diverse range of use cases across different fields introduces unique characteristics and challenges that further complicate the forecasting process (Allende and Valle 2016).

Driven by advances in both statistical methods and computational power, the field of time series forecasting has evolved significantly over the past century. One of the first milestones was achieved when Yule 1927 introduced the concept of stochasticity in time series. Building on this foundation, researchers such as Slutsky, Walker, Yaglom, and Yule formulated the concepts of autoregressive (AR) and moving average (MA) models, which enabled the modeling of dependencies in sequential data (De Gooijer and R. J. Hyndman 2006).

In the 1950s and 1960s, exponential smoothing methods emerged as a popular approach for time series forecasting. These methods, introduced by Brown 1959, Healy and Brown 1964, Holt 2004, and Winters 1960, gained popularity for their simplicity and effectiveness in handling trends and seasonality. Exponential smoothing assigns exponentially decreasing weights to past observations, emphasizing more recent data, and is often used to complement other forecasting techniques (Brown 1959, Healy and Brown 1964, Holt 2004, Winters 1960).

In 1970, Box and Jenkins developed the Autoregressive Integrated Moving Average (ARIMA) model, marking a significant breakthrough in time series forecasting. ARIMA provided a framework for modeling non-stationary time series through differencing and the combination of AR and MA components (Box et al. 2016). This model specifically addresses the issue of non-stationarity by incorporating a differencing term, which represents the number of differencing operations required to achieve stationarity, thereby simplifying the complexities within time series data (Shumway and Stoffer 2017). To specifically address seasonal variations, the Seasonal Autoregressive Integrated Moving Average (SARIMA) model was subsequently developed, extending the ARIMA framework by incorporating seasonal differencing as well as seasonal autoregressive and seasonal moving average components (Box et al. 2016, Chapter 9).

Building on the foundational advancements of ARIMA and SARIMA, the integration of external information further enhanced the flexibility of time series models. The development of the Seasonal Autoregressive Integrated Moving Average with Exogenous Variables (SARIMAX) model extended the SARIMA framework by allowing the inclusion of external factors that influence the time series (Korstanje 2021a). By accounting for these exogenous influences, SARIMAX provided a versatile tool for handling complex, real-world data. Vagropoulos et al. 2016 demonstrated that SARIMAX can outperform SARIMA in day-ahead forecasting, showcasing its competitiveness with other established methods.

The late 20th century saw the emergence of machine learning models for time series forecasting, introducing a new paradigm. Ensemble methods gained prominence due to their ability to model non-linear relationships (Connor, Martin, and Atlas 1994, Chakraborty et al. 1992). They combine the predictions of multiple weak learners to create a more accurate model (Dietterich 2000). Extreme Gradient Boosting (XGBoost), introduced by Chen and Guestrin 2016 is one example of this class of models, which iteratively refines predictions through weighted corrections of prior errors. The incorporation of regularization mechanisms helps mitigate overfitting, making it particularly effective in complex forecasting tasks (Bühlmann 2012).

In parallel, advancements in deep learning have further expanded the forecasting toolkit. Long Short-Term Memory (LSTM) networks, introduced by Hochreiter and Schmidhuber 1997, addressed key limitations of traditional neural networks in handling sequential data. By capturing long-term dependencies in time series, LSTM has become a cornerstone of modern deep learning applications in forecasting. These models have significantly advanced the field, enabling robust handling of complex temporal patterns and non-linear relationships (LeCun, Bengio, and Hinton 2015).

In the early 2000s, the Theta model gained recognition for its simplicity and robustness (Vassilis Assimakopoulos and Nikolopoulos 2000). This model was particularly highlighted during the M3-competition, a renowned forecasting competition aimed at evaluating the accuracy and utility of various time series forecasting methods (De Gooijer and R. J. Hyndman 2006, Chapter 2.2). This model decomposes de-seasonalized data into two ”Theta lines”: the first emphasizes

the long-term trend by completely smoothing the curvature, while the second amplifies curvature to capture short-term dynamics. These components are recombined to create a robust forecast that balances long-term and short-term patterns effectively. The model is particularly notable for its simplicity and adaptability across different time series characteristics (Vassilis Assimakopoulos and Nikolopoulos 2000).

More recently, in 2017, Facebook's Data Science Team introduced the Prophet model (S. J. Taylor and Letham 2018). Designed for practical use, Prophet employs a decomposable framework incorporating trend, seasonality, and holidays, making it particularly effective for time series with irregularities such as missing data or outliers (Vishwas and Patel 2020). With its intuitive interface and flexibility, Prophet allows users to adjust parameters without requiring an extensive understanding of the underlying model (S. J. Taylor and Letham 2018). Its multivariate capabilities also enable the simultaneous analysis of multiple variables (F. Wang and Aviles 2023) and the incorporation of exogenous variables (Shumway and Stoffer 2000), enhancing its predictive power in complex scenarios (R. Hyndman and Athanasopoulos 2012).

1.4.2 Time Series Forecasting at Call Centers

Given its versatility, time series forecasting has been extensively applied to predict call center arrivals, where accurate predictions are vital for optimizing staffing levels and managing resources efficiently. Early studies primarily employed classical statistical models, such as SARIMA and exponential smoothing methods, which can capture basic trends and seasonality in call arrival data (J. W. Taylor 2007, Petropoulos et al. 2022, Chapter 3.8.4.). However, the evolution of forecasting mirrors broader trends in the field, shifting towards more advanced machine learning techniques. Particularly, deep learning models, such as LSTM and its variants have shown promise in handling large-scale call center datasets and complex temporal dependencies, providing more robust predictions (Torres et al. 2021, Lim and Zohren 2021).

1.4.3 Time Series Forecasting in Limited Size Datasets

In both the general introduction of time series forecasting and the more specific discussion on its application to predicting call center arrivals, a clear trend emerges: the increasing reliance

on neural networks and deep learning models.

As highlighted by LeCun, Bengio, and Hinton 2015, neural networks, however, require large datasets to train effectively because they need a substantial amount of data to properly adjust their numerous parameters and avoid overfitting. "Overfitting occurs when the gap between the training error and test error is too large" (Goodfellow, Bengio, and Courville 2016, Chapter 5, p. 110) implying that the model learns not only the underlying patterns in the data but also the noise, which leads to poor generalization to new, unseen data. In the case of neural networks, this issue is particularly pronounced because these models have a large number of parameters that need to be adjusted. Moreover, pretrained neural networks, despite containing a significant number of non-learning neurons, face similar challenges, as they can still struggle with overfitting when insufficient data is provided.

1.5 Project-Based Learning Project

To address the challenge of predicting the optimal volume of customers to flag each day for the Detractor Squad as introduced in 1.2 (Context), the Project-Based Learning (PBL) program was undertaken. PBL is a pedagogical approach in which students are actively engaged in real-world projects, applying their knowledge and skills to solve concrete problems. It offers a hands-on learning experience that bridges the gap between theory and practice. Through the PBL program, the opportunity was provided to delve deeper into the second stage of NOS' two-stage approach: the time series forecasting model used to determine the daily number of customers to flag. Specifically, efforts were focused creating a model that outperforms the ARIMA $(1, 0, 1)$ model currently implemented by NOS for this purpose, on the time series data at hand. This section presents a detailed overview of the PBL project and its outcomes.

1.5.1 Target Variable and Data Preprocessing

As outlined in section 1.2 (Context), the objective of the PBL project was to predict the optimal volume of customers to flag each day for the Detractor Squad. However, the dataset, as detailed in section 1.3 (Dataset), was structured at the client level rather than daily granularity, resulting in the absence of a suitable target variable. Consequently, the initial step within the PBL pro-

gram involved the transformation of the dataset into daily granularity to establish an appropriate target variable. Considering that the Detractor Squad exclusively manages inbound calls, which are calls initiated by clients, the dataset was filtered to eliminate any records of outbound calls, i.e. calls initiated by NOS' call center. Furthermore, since the Detractor Squad does not operate on weekends, any records from weekends were also excluded from further analyses. The resulting target variable, representing the number of people to flag in order to meet the daily target of actual calls for the Detractor Squad, is shown in figure 1.

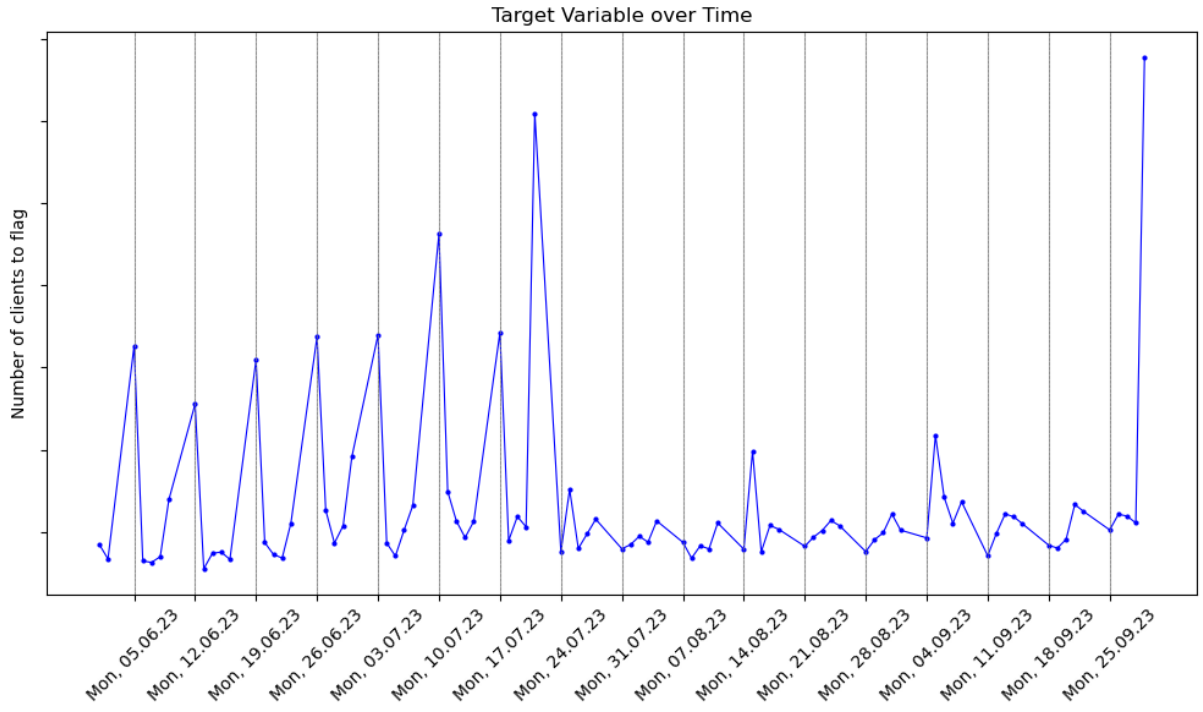


Figure 1: The target variable [#clients to flag] plotted over time

The x-axis represents the time period from May 31st to September 29th, 2023, while the y-axis shows the number of clients flagged. From the start of the dataset until late July, the figure displays a spiky pattern. Notably, during this period, clear seasonality was observed since spikes predominantly occurred on Mondays. Nevertheless, the final spike from that period deviated from this pattern by occurring on a Tuesday. Through close collaboration with NOS, these extreme spikes were identified as outliers in the target variable.

Following the initial spiky phase, a more stable pattern emerged from late July through the end of September. This stability, punctuated by only a couple of smaller spikes, all occurring on a Tuesday, was, according to NOS, more representative of the overall behavior of the data. A no-

table feature of this dataset was the massive spike in the target variable at the end of the provided time series. This abrupt increase could signify a seasonal shift, a change in client behavior, or an external event impacting the number of clients needing attention. Similar to the extreme spikes observed during the previously mentioned spiky period, this spike was considered an outlier, confirmed by NOS.

1.5.2 Feature Engineering

Feature engineering was conducted following the same logic as the computation of the target variable, with daily-grouped features being created from the original, more granular data. Since the objective was to predict the number of clients to flag each day, variables that could impact call volume and behavior were prioritized. The business expertise provided by NOS was invaluable in guiding decisions regarding the selection of variables.

The features deemed most important included call duration, hold time, total number of calls to the technical line, and client score. Call duration referred to the length of each interaction, while hold time indicated the duration clients spend waiting for assistance. The client score, as already outlined in section 1.3 (Dataset), reflected the likelihood of a client calling within the next 15 days. Additionally, information extracted from the date, such as the weekday and month, were also considered significant.

Importantly, rather than relying on a single metric for these variables, their daily sum, mean, and standard deviation were employed, providing a more comprehensive representation of daily patterns in the dataset.

1.5.3 Methodology

1.5.3.1 Target Capping

As mentioned in section 1.5.1 (Target Variable and Data Preprocessing) and evident in figure 1, the target variable exhibits both numerous and extreme outliers. This resulted in a spiky pattern from the start of the dataset until late July, along with a significant spike at the end of the dataset. The high quantity of extreme outliers was observed to distort the model's understanding

of typical patterns, leading to inaccurate predictions and an inability to generalize effectively to the underlying data distribution. Consequently, strategies such as capping the target variable, were implemented to mitigate the adverse effects of outliers and enhance overall predictive accuracy.

Capping started by identifying these extreme values using an Interquartile Range (IQR) technique (John W. Tukey 1977) with a multiplier of 1.5. The IQR represents the central 50% of the data, i.e. the range between the first and third quantiles. Any point outside 1.5 times the IQR above the third quantile or below the first quantile was considered as an outlier.

After identifying the outliers, these extreme values were adjusted to the maximum value in the non-outlier data. However, the difference between the original outliers and the capped values was not entirely discarded. Instead, the difference was scaled down by 20%, preserving some of the relative size of these spikes to ensure that the pattern of variability in the data was maintained while diminishing their impact.

The chosen hyperparameters for IQR multiplication and scaled difference resulted from simple experimentation with various combinations to identify the optimal configuration. Figure 2 illustrates the impact of capping on the operational target. The blue line represents the original operational target, whereas the red line indicates the capped version. The capping effectively reduced the magnitude of extreme spikes while preserving the overall structure and trends of the data, ensuring a more stable and practical input for forecasting models.

1.5.3.2 Introduction of the Models Deployed

During the project, a structured three-step approach was employed, progressing from baseline models to univariate models, and finally to multivariate models. This systematic progression enabled an incremental increase in complexity.

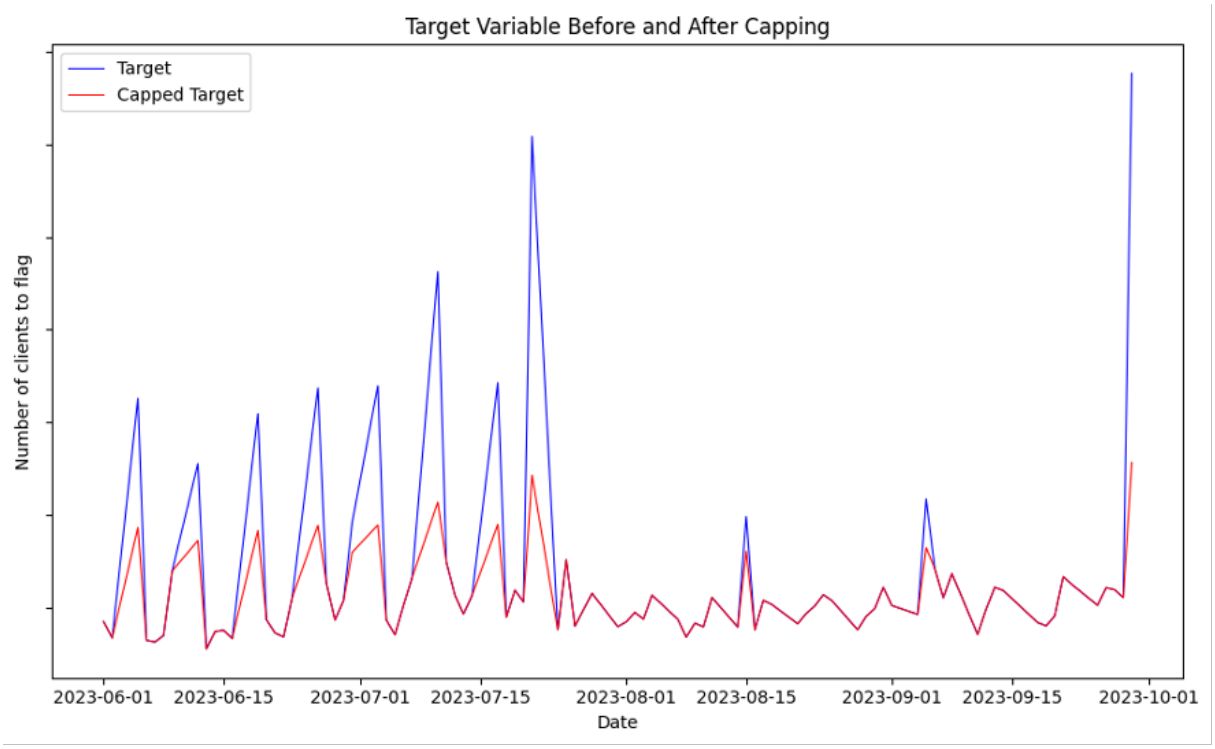


Figure 2: Impact of the capping on the target variable

1.5.3.2.1 Limitations due to Small Dataset

As discussed in section 1.3 (Dataset) the time series available spanned from May 31, 2023, to September 29, 2023, covering only 122 days. Once weekends were excluded, as mentioned in section 1.5.1 (Target Variable and Data Preprocessing), the dataset was further reduced to 88 days. This posed a challenge for the analysis since time series are characterized by recurring seasonal trends (R. J. Hyndman and Athanasopoulos 2018a). With only four months of data and given the daily granularity, complete seasonality was absent, and the diversity of observations was limited, making it difficult to capture these trends and generate accurate future predictions. Additionally, the small dataset increased the model's sensitivity to outliers, complicating the distinction between significant signals and noise. This further encouraged overfitting, which compromised the model's ability to generalize to new data (Shumway and Stoffer 2000).

Moreover, as highlighted in section 1.4.3 (Time Series Forecasting in Limited Size Datasets), neural networks, while powerful, require large datasets to avoid issues such as overfitting, where the model learns the noise rather than the underlying patterns. Given these constraints and the risk of poor generalization with small datasets, the focus during the PBL program was

not placed on complex neural networks but rather on simpler models with fewer, more easily tunable parameters. This approach was adopted to mitigate the risks associated with limited data and to develop models that were more stable and reliable for the analysis.

1.5.3.2.2 Baseline Models

The initial focus was placed on establishing baseline models that served as an essential point of reference for the performance of subsequent, more sophisticated models. The first baseline model predicted the next value by using the average of data from a fixed window of past time points.

As discussed in section 1.5.1 (Target Variable and Data Preprocessing), consistent patterns in the data corresponding to specific weekdays were identified. This observation led to the implementation of the Weekday Average model, which was designed to capture these recurring patterns. While effective at highlighting significant peaks, this model also revealed the challenges posed by abrupt changes in the data, such as the spiky pattern from late May to late July, followed by a period of stability extending through September. The model's sensitivity to such fluctuations emphasized the need to evaluate the stability and consistency of predictions, in addition to traditional error-based metrics. This dual focus on correctness and consistency, which is further explored in section 1.5.3.3 (Model Evaluation Criteria) became a critical aspect of the modeling approach.

In addition to these models, ARIMA $(1,0,1)$ was incorporated, recognizing its significance as the model currently employed by NOS. By comparing the results with this model, potential advancements in the performance of the attempted approaches could be highlighted. Thus, this model served as the most important benchmark throughout the project and was referred to as the "NOS Benchmark" throughout the project.

Throughout the baseline phase, an embed size of 30 was consistently utilized, aligning with the configuration currently used by NOS. This embed size defined the maximum training window for each iteration of the temporal cross-validation (TCV), where the training set iteratively grew each day until it reached the predetermined embed size and then remained fixed. A more

detailed explanation of the TCV approach is provided in section 1.5.3.3.4 (Temporal Cross Validation).

1.5.3.2.3 Univariate Models

In the second phase, univariate time series analysis was conducted, and several prominent models were applied to capture different patterns and trends in the data.

The univariate analysis was initiated with the ARIMA model, as this model specifically addressed the issue of non-stationarity by incorporating a differencing term, as mentioned in section 1.4.1 (Time Series Forecasting). Given the pronounced fluctuations and the absence of a constant mean or variance in the dataset, the differencing component was pivotal in addressing these irregularities.

The limitation of having only 88 days of data, constrained the ability to identify long-term trends and seasonal effects. Despite this limitation, ARIMA models are generally still effective in capturing short-term dependencies (Shmueli and Polak 2024, Box et al. 2016) aligning with the objective of predicting two workdays ahead.

While ARIMA is generally effective for non-seasonal data, the SARIMA model was adopted to enhance the analysis due to the periodic patterns observed in the dataset, outlined in section 1.5.1 (Target Variable and Data Preprocessing). By addressing these periodic patterns, SARIMA proved valuable for datasets with recurring fluctuations (Box et al. 2016), such as the ones at hand. The need to account for the cyclical behaviors, which motivated the introduction of the Weekday Average model, as discussed in section 1.5.3.2.2 (Baseline Models), highlighted the importance of accurately modeling consistent patterns over time. This approach allowed seasonality to be explicitly accounted for and the periodic patterns, critical for accurately modeling consistent behaviors, to be captured.

To complement the SARIMA model, Exponential Smoothing was also applied, leveraging its adaptability to respond effectively to shifts in data patterns. This approach ensured sensitivity to short-term changes in the time series, enhancing forecast accuracy.

The last univariate model that was implemented was the Theta model, due to its adaptability across different time series characteristics (Vassilis Assimakopoulos and Nikolopoulos 2000). Notably, this model was also employed at NOS prior to their adoption of the ARIMA $(1,0,1)$ model, thereby providing a compelling rationale for its implementation in this comparative analysis.

1.5.3.2.4 Multivariate Models

Besides tackling univariate modeling, the project also involved exploring multivariate models to capture potential relationships within the data that simpler univariate models could have missed.

The first multivariate model deployed was a multivariate extension of the SARIMA model, SARIMAX, as it is a natural progression from the univariate econometric models ARIMA and SARIMA, discussed in the previous section. Furthermore, this model offers greater flexibility when dealing with complex time series data, which was particularly relevant given the nature of the time series (Korstanje 2021a).

Following SARIMAX and continuing along the multivariate time series modeling trajectory, Prophet was selected due to its capability to handle data with various components such as trend, seasonality, and holiday effects (S. J. Taylor and Letham 2018). This feature is particularly advantageous given the cyclical nature of the dataset. Additionally, the model is known for efficiently handling outliers and offering fast model fitting (Vishwas and Patel 2020), making it a suitable choice for addressing the spiky patterns present in the data and the need for daily forecasts.

After exploring time series models such as SARIMAX and Prophet, a transition was made to machine learning approaches to explore an alternative technique and potentially achieve improved results. The final multivariate model considered was XGBoost, chosen for its ability to capture the complexity of the data (Y. Wang and Guo 2020). Moreover, it effectively mitigates overfitting through regularization (Chen and Guestrin 2016), addressing the generalization preferences highlighted by NOS.

1.5.3.3 Model Evaluation Criteria

An inappropriate choice of evaluation metrics can adversely impact the entire model lifecycle, from development to deployment. As mentioned in section 1.5.3.2.2 (Baseline Models), the importance of a dual approach that balanced both performance and consistency was demanded by NOS. From a performance perspective, the model was required to accurately predict the daily call volume for the Detractor Squad, ensuring that relapsing clients received adequate attention. On the consistency side, the model needed to deliver stable predictions that minimized day-to-day fluctuations in call volume, supporting a balanced workload for the Detractor Squad and ensuring operational efficiency. This dual focus ensured that both the immediate demands of relapsing clients and the long-term sustainability of team performance were addressed.

1.5.3.3.1 Measurement Variable

The target variable, which represented the number of clients to flag each day, did not align well with the operational reality of the Detractor Squad. Since their workload was determined by the number of calls they handled daily, using flagged clients also as a measurement variable, i.e., the variable on which the evaluation metrics are based, became impractical. Given that not all flagged clients would initiate calls, relying on flagged client predictions led to significant discrepancies in the resultant call volume. A model might indicate a need to flag many clients, but this does not necessarily correlate to a similar increase in the actual number of calls received. This inconsistency complicated the evaluation of the model's true effect on the Detractor Squad. To address this, a new measurement variable was introduced, representing the number of daily calls resulting from flagging a specific number of clients. To determine the number of clients that should have been flagged to align with operational requirements, the daily ordered list of client scores was used, along with the column indicating whether each client made a call on that specific day. The list of clients who made a call was then progressed through until the desired call volume was reached, thereby meeting the operational needs. The same approach was applied to calculate how many calls would have resulted from the model's predictions. This adjustment provided a more relevant and practical assessment of the model's performance, as it

allowed for direct comparison with the operational call volume requirements.

It is important to clarify that the target variable remained the number of clients to flag each day. The daily call volume requirement for the Detractor Squad solely served as a measurement variable for evaluating the model's performance. Therefore, while the models were required to predict the number of clients to flag, the error and smoothness metrics were calculated based on the actual number of calls received, reflecting a more meaningful assessment of model performance. These metrics will be explored in the subsequent section.

1.5.3.3.2 Evaluation Metrics

To evaluate the models, two distinct metrics were employed in line with the dual approach outlined in section 1.5.3.3 (Model Evaluation Criteria). To assess performance, the Mean Absolute Percentage Error (MAPE) (Korstanje 2021b) was selected. This metric was chosen due to its ability to present errors in relative terms rather than absolute values, enabling consistency across varying data scales and ensuring reliable comparative evaluations. Moreover, MAPE was already in use by NOS for this particular problem, aligning the approach with existing practices.

The Smoothness Index (SI) was selected as the metric to assess stability. Defined as the Mean Squared Differences divided by the Variance of the observations, the SI measures the variability between predictions. This metric, developed by the PBL team and validated in collaboration with NOS, was used to track the model's stability and consistency throughout the forecasting process. Monitoring the SI ensured that the Detractor Squad's workload remained relatively consistent, minimizing significant fluctuations in day-to-day operations.

1.5.3.3.3 Best Case Scenario

As discussed in section 1.2 (Context), overestimating call volumes was prioritized by NOS rather than underestimating them, given the higher cost associated with idle time in the Detractor Squad. To reflect this preference, a tolerance of 50 calls above the daily operational target was established in collaboration with NOS, creating a target-tolerance band, as shown in figure 3.

Any forecast that fell within this specified range was deemed to be ideal, representing what was referred to as the "best-case scenario". If forecasts fell just below the target by one call or exceeded the upper tolerance threshold by one call, both instances were considered equally distant from the best-case scenario, underscoring the emphasis on overestimation.

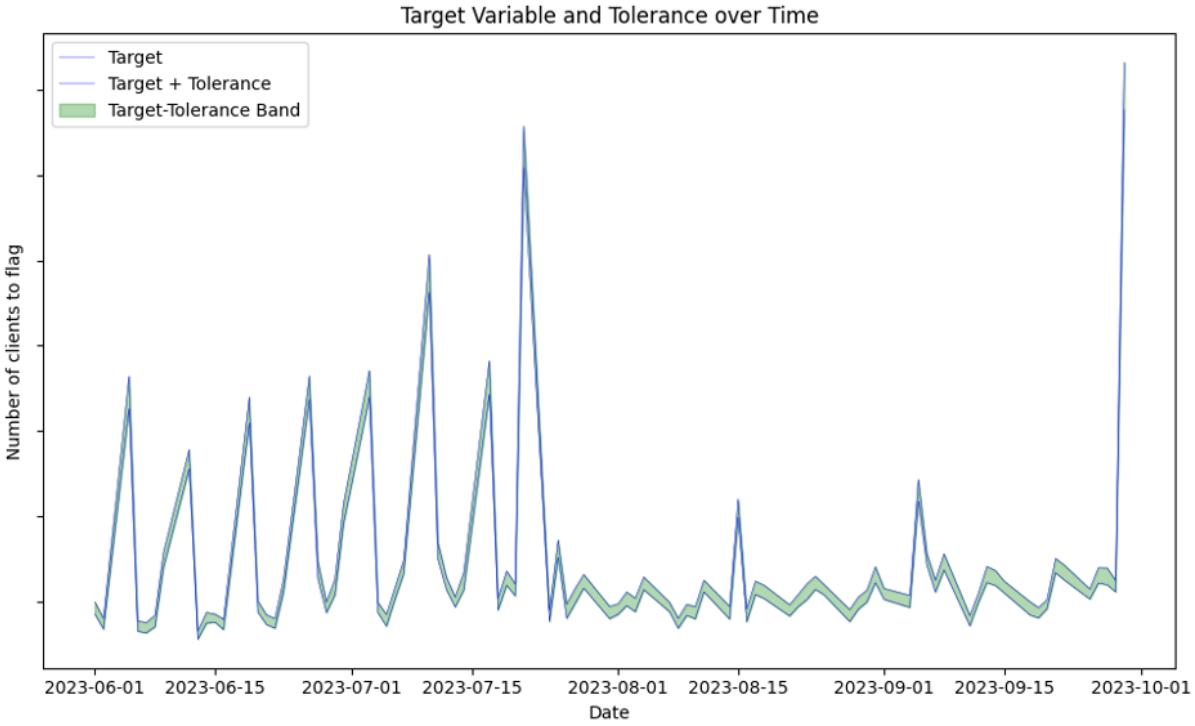


Figure 3: Target-Tolerance Band

This approach not only aligned with the operational preferences of NOS but also provided a structured framework for evaluating the performance of the forecasting models. By establishing a clear definition of what constituted an optimal outcome, the degree to which each developed model approached the best case scenario could be assessed. Consequently, the results presented in this study consistently indicate the degree to which each model deviated from the "best-case scenario".

1.5.3.3.4 Temporal Cross Validation

To evaluate the effectiveness of a predictive model, it was crucial to ensure that the model performed well not only on the data used for training but also on new, unseen data. To achieve this, a common approach involved splitting the available data into separate subsets, allowing

the model to be trained on one part while being tested on another. This idea formed the basis of what is known as cross-validation (Berrar et al. 2019), a technique where the data is divided into multiple subsets, or folds, and the model is trained and tested across different combinations of these subsets.

In time series data, cross-validation is not appropriate because the data has a natural time order (Cerqueira, Torgo, and Mozetič 2020, Bergmeir and Benítez 2012, Falessi et al. 2020). Randomly splitting the data for cross-validation could produce misleading results, as training on future data and testing on past data could lead to data leakage and compromise the evaluation's validity. Instead, techniques like forward validation, introduced by Hjorth 1994, were used to respect the time order by only using past data to predict future outcomes.

The core concept of these techniques revolve around the rolling window approach. This method begins by selecting an initial subset of data and then progressively shifts the window forward with each prediction iteration. It is crucial to note that a predetermined embed size, established as a hyperparameter, must be defined before implementing this method (Bergmeir and Benítez 2012).

By training on successive subsets of data rather than the entire dataset, the model was able to adapt to evolving temporal patterns, hence mitigating the risk of overfitting (Berrar et al. 2019), a significant concern given the size of the dataset.

Therefore, to validate the models, temporal cross-validation (TCV) was developed, drawing inspiration from forward validation. In TCV, an initial split was made where the training set comprised data from one day, and the test set consisted of data from two days ahead, aligning with NOS' request for predictions to be made two days in advance, as explained in section 1.2 (Context).

As the process progressed, each new split incorporated the previous test set into the training set, which then expanded sequentially by one day. Once the training set reached the predetermined embed size, the process continued by rolling the window forward. A visualization of the TVC, given an embed size of 15 is shown in figure 4 below.

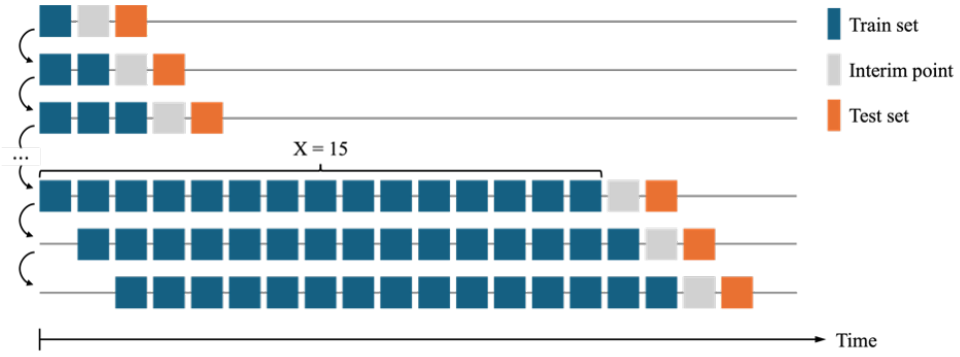


Figure 4: Visualization of the Temporal Cross Validation given an embed size of 15

1.5.3.3.5 Evaluation Window

The implementation of temporal cross validation introduced a new challenge. While optimal performance was achieved by various models with different embed sizes, this variation in window size led to an unequal number of predicted data points across models.While various models may achieve optimal performance with different embed sizes, this variation in embed size could lead to an unequal number of predicted data points across models. Consequently, when comparing model performance, this discrepancy could introduce bias, potentially skewing the evaluation results.

To address this issue, model performance was tracked for the last 40 days based on two key factors. First, all models were developed using an embedding size of 30 days or less due to the dataset's limited size, which prevented capturing meaningful seasonality patterns with larger embedding sizes. Second, the initial period from May to July exhibited a spiky pattern, as noted in section 1.5.1 (Target Variable and Data Preprocessing), while the last 40 days provided a more representative view of typical operational patterns, a characteristic confirmed by NOS.

Furthermore, the final day was excluded from the metrics analysis, as it was considered an outlier that could have disproportionately affected the metrics and led to misleading results, as discussed in section 1.5.3.1 (Target Capping). This approach ensured that all models were evaluated on 39 predictions, allowing for fair and unbiased comparisons while mitigating the impact of outliers.

1.5.3.3.6 Tailored hyperparameter tuning

To optimize the performance of the models, grid search was employed, a hyperparameter tuning method widely used in data science. This approach exhaustively explored a pre-defined subset of hyperparameter combinations for the model it was applied to. By evaluating every possible configuration, grid search determined the optimal parameters that delivered the best results according to a specific metric.

In this case project, a customized version of grid search was required that could first accommodate variable parameters and embed sizes, and secondly, allow comparisons of different metrics, aligning with the aforementioned dual approach. As mentioned in the previous section, outliers were consistently excluded from the metric calculations to ensure the integrity of the MAPE results.

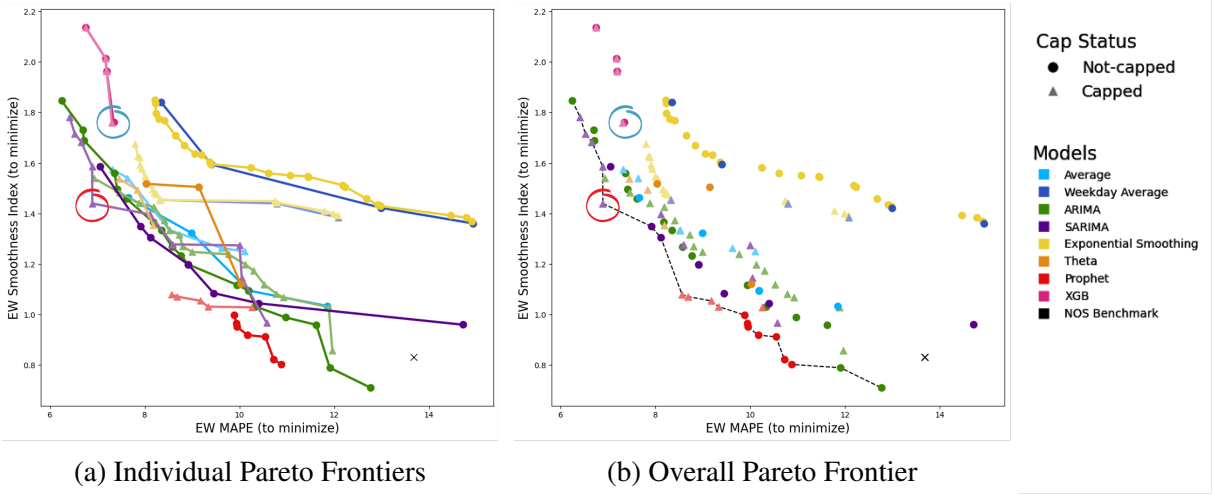
In conjunction with grid search, the multi-objective optimization problem was tackled by employing the concept of Pareto Efficiency, named after the Italian economist Vilfredo Pareto (Ehrgott 2012, Pindyck et al. 2018, Chapter 16.2). A model's outcome is Pareto efficient if any possible reconfiguration of the hyperparameters would lead to a deterioration in at least one metric (Yao et al. 2023). In this context, reducing MAPE required adjustments of the hyperparameter configuration, which, in turn, would have led to a less favorable SI, and vice versa. All these Pareto efficient configurations collectively formed the so-called Pareto Frontier. By focusing on this frontier, non-Pareto efficient points that did not contribute to the goal of achieving an optimal balance between MAPE and SI were disregarded.

This approach enabled the identification of the most promising hyperparameter configurations while also providing a deeper understanding of the trade-offs inherent in the model's performance.

1.5.4 Results from PBL

In this analysis, the performance of the forecasting models was evaluated by comparing them to the established benchmark, providing a clear assessment of their effectiveness and potential impact on NOS' operations. To guide model selection, Pareto frontiers were created for each

model, highlighting configurations that achieved the optimal balance between MAPE and SI.



(a) Individual Pareto Frontiers (b) Overall Pareto Frontier

Figure 5: Pareto frontier PBL

Figure 5 provides a visual comparison of all algorithms and their respective tested configurations through the application of the Pareto-frontier concept, as introduced in section 1.5.3.3.6 (Tailored hyperparameter tuning). Figure 5a illustrates the Pareto-efficient configurations within each algorithm, while figure 5b presents the overall Pareto-frontier (OPF), indicated by the black dashed line, encompassing all configurations tested. In both panels, each algorithm was distinguished by a unique color. To indicate whether configurations incorporated the target capping concept, introduced in section 1.5.3.1 (Target Capping), different marker shapes were employed where circles denoted configurations based on non-capped data, whereas triangles represented configurations based on capped data.

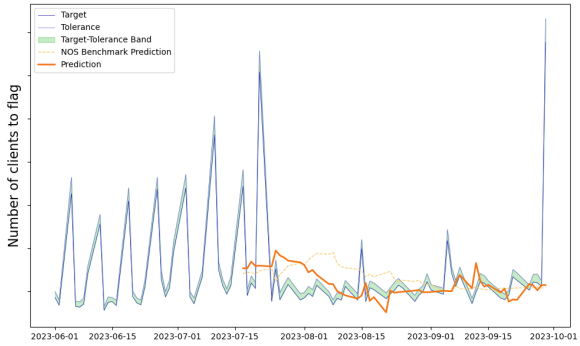
In general, figure 5 demonstrated that increased stability often came at the expense of predictive power. Along the OPF, starting from the top, econometric univariate models such as ARIMA and SARIMA dominated, indicating that these models performed well in terms of forecast precision, achieving lower MAPE values, but exhibited high instability. Further down the OPF, the SARIMA model continued to appear, reflecting its ability to achieve strong results with low MAPE values while outperforming the majority of other algorithms in terms of stability with

lower SI values. Toward the lower end of the OPF, Prophet models were followed by ARIMA models, which demonstrated the best performance in terms of stability but tended to sacrifice precision, as reflected in higher MAPE values.

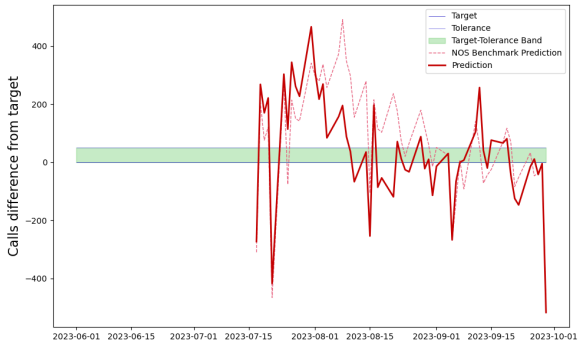
Notably, the NOS Benchmark model, represented by the black x and positioned in the low SI and high MAPE region, did not appear on the OPF. This absence suggested that, compared to the other models, it failed to achieve the optimal balance between MAPE and SI. Consequently, selecting any model from the OPF offered a more effective solution for the forecasting task related to call center arrival predictions than the benchmark currently utilized by NOS.

1.5.4.1 Deployable Solution

In the pursuit of identifying the most practical model for deployment, it was essential to select a configuration that achieved the optimal balance between MAPE and SI. The SARIMA $[(2,0,1),(0,0,1,20)]$ configuration with a 20-day embed size, combined with the target capping method introduced in section 1.5.3.1 (Target Capping), is illustrated by the red-outlined point in figure 5. From here on referred to as the deployable solution, this configuration delivered a MAPE of 6.89 with an SI of 1.44. Compared to the NOS Benchmark, with a MAPE of 13.68 and SI of 0.83, this represented a 49.6% reduction and a 73.5% increase, respectively. Although the increase in SI appeared considerable, the substantial reduction in error reflects a more favorable trade-off for the application at hand, prioritizing predictive accuracy over stability.



(a) Target variable (#people to flag) over time with respective predictions



(b) Measurement Variable (#calls) over time with respective predictions

Figure 6: Performance of the Deployable Solution - PBL

Figure 6 visualizes the performance of the deployable solution. As established in section 1.5.1 (Target Variable and Data Preprocessing), the primary challenge for this forecasting task, given the dataset, was the abrupt transition in patterns from spiky to non-spiky.

Figure 6a illustrates the performance of the NOS Baseline model and the deployable solution on the target variable, defined in section 1.5.1 (Target Variable and Data Preprocessing) as the number of people to flag. During the transitional period of the time series, the deployable solution demonstrated a faster adaptation to the non-spiky pattern compared to the NOS Benchmark model. Further, 6b, which illustrates the performance on the measurement variable, defined in section 1.5.3.3.1 (Measurement Variable) as the number of calls, shows that the deployable solution's predictions were closer to the target-tolerance of the measurement variable in most cases, which was quantified in the reduction in MAPE.

In conclusion, this SARIMA configuration provided a well-balanced trade-off between minimizing error and ensuring a manageable level of forecast variability.

1.5.4.2 Build-Onto Solution

As established, the deployable solution offered the best trade-off between MAPE and SI on the given dataset. However, the limitation regarding dataset size made a full optimization of the multivariate models infeasible. Therefore, due to the significance of this issue for call center operations, as discussed in section 1.2 (Context), a multivariate "build-onto" solution was provided alongside delivering the deployable solution, intended to serve as a foundation for further refinement and optimization if additional data becomes available.

Among the multivariate models evaluated, the XGBoost model demonstrated the lowest MAPE values observed throughout the project. The chosen configuration of the XGBoost algorithm is represented by the blue-outlined point in figure 5. This model employed 35 estimators, an absolute error regression objective, a maximum depth of 25, an L1 regularization term of 0.1, and a 10-day embed size. It achieved a MAPE value of 7.5, representing a reduction in error of more than 45% compared to the NOS Benchmark, highlighting its strong predictive accuracy. However, the XGBoost configuration recorded an SI value of 1.76, one of the worst perfor-

mances regarding smoothness, indicating its inability to achieve a balance between accuracy and stability. This limitation also explained why it did not appear on the OPF.

Nevertheless, this model, being multivariate, can capture complex interactions in variables, as mentioned in section 1.5.3.2.4 (Multivariate Models). As such, it has much latent potential for further optimization and further exploration of performance with more datapoints, as well as with more aggregate variables from other sources.

2 Enhancing the performance of time series models through Unsupervised Outlier Detection and Treatment Techniques

The presence of outliers has long been recognized as a challenge in data analysis, as they can distort results and reduce the reliability of statistical models (D. Montgomery and Runger 2010). These outliers are often defined as observations that significantly differ from the majority of data points, arising from measurement errors, data entry mistakes, or rare but valid events within the data-generating process (Aggarwal 2016). Due to their deviation from expected patterns, outliers can influence model outcomes, potentially leading to erroneous conclusions (Rousseeuw and Leroy 1987). Over time, a variety of methods have been developed to identify and handle outliers, with the objective of either mitigating their influence or uncovering the unique conditions they represent (Barnett and Lewis 1984).

Effectively addressing outliers is essential for building robust predictive models. When untreated, outliers can introduce bias, overemphasize rare events, or mask meaningful trends within the data, all of which can reduce the predictive power and robustness of the models (Hodge and Austin 2004). Outlier treatment helps stabilize models by ensuring that extreme values do not disproportionately influence predictions, resulting in outputs that better reflect general patterns (Aggarwal 2016). However, since outliers are not always erroneous, sometimes representing legitimate but unusual data points that reveal important insights (Zimek, Schubert, and Hans-Peter Kriegel 2012), a careful evaluation is required to determine whether these data points should be excluded or leveraged to enhance understanding of the underlying phenomena

(Rousseeuw and Hubert 2017).

In the initial PBL phase, the challenges posed by outliers in the dataset were overcome by deploying a capping strategy applied to the target variable. This approach involved applying the Interquartile Range (IQR) technique to identify extreme values and adjusting them by capping and scaling the difference to preserve variability. While this method effectively reduced the influence of extreme outliers and ensured a more stable input for modeling, it was limited in scope, as it addressed only the target variable and did not account for potential distortions in other features. A detailed explanation of the capping procedure implemented during this phase is provided in section 1.5.3.1 (Target Capping).

Building on the insights gained during the PBL phase, this individual work addresses these limitations by introducing more advanced and comprehensive outlier detection and treatment methodologies. By expanding the scope to include additional features and employing techniques such as Local Outlier Factor (LOF), Isolation Forest, and density-based methods like DBSCAN, this thesis seeks to create a robust framework for outlier management. The proposed approach is designed to improve predictive accuracy and model stability while maintaining the integrity of significant patterns in the data.

2.1 Literature review

This chapter provides an overview of the concepts and methodologies used in this thesis. It begins with a discussion on the definition of outliers, followed by an exploration of unsupervised outlier detection techniques and the role of normalization in improving their effectiveness. The chapter then examines approaches for addressing identified outliers, focusing on removal and modification strategies. In addition, robust estimation techniques, which reduce the impact of outliers without explicitly identifying or altering them, are analyzed. The chapter concludes with a discussion of the implications of outlier treatment in both the target variable and the features.

2.1.1 Definitions of outliers

The presence of outliers has been the subject of extensive research across various fields resulting in evolving definitions and interpretations. Grubbs 1969 described an outlier as "an observation that appeared to deviate markedly from other members of the sample in which it occurred". Hawkins 1980 expanded on this concept, raising the "suspicion that an outlier was generated by a different mechanism". Barnett and Lewis 1984 defined outliers as "observations which appeared to be inconsistent with the remainder of that set of data". Muñoz-Garcia, Moreno-Rebollo, and Pascual-Acosta 1990 attempted to avoid the subjective side of these definitions and added the condition that the observation should deviate decidedly from the general behavior "in relation to the criterion on which the analysis was carried out". Finally, Papadimitriou et al. 2003 highlighted the distinct characteristics of outliers compared to their neighbors.

In summary, while the definitions of outliers vary, a common thread persists: outliers are data points that deviate substantially from the rest of the dataset, prompting further investigation into their origins and implications.

2.1.2 Unsupervised outliers' detection techniques

Outlier detection involves identifying irregular data points that deviate significantly from the general pattern (Duan et al. 2008). In cases where labeled data, indicating which points are outliers, is unavailable, unsupervised outlier detection techniques are particularly valuable, as they rely solely on the inherent structure and distribution of the data to identify outliers (Hinton and Sejnowski 1999). This approach is particularly relevant for NOS, as the organization does not have alarmist systems or predefined labels that indicate which data points are outliers. Consequently, the lack of labeled data necessitates the use of unsupervised methods. These techniques are commonly grouped into statistical methods, clustering-based methods, nearest-neighbors methods, and machine learning-based methods.

Statistical methods for outlier detection assume that "normal data instances occur in high-probability regions of a stochastic model, while outliers occur in the low-probability regions of the stochastic model" (Hodge and Austin 2004). A frequent assumption in these methods is that

the data follows a normal distribution, as this allows for straightforward identification of deviations (Barnett and Lewis 1984). When this assumption is not met, transforming non-normally distributed variables can improve the effectiveness of outlier detection by reducing skewness and approximating normality (Wackerly, Mendenhall, and Scheaffer 2014). Techniques such as log (Wackerly, Mendenhall, and Scheaffer 2014), square root (D. C. Montgomery, Jennings, and Kulahci 2015), and Box-Cox transformations (R. J. Hyndman and Athanasopoulos 2018b) are commonly employed to stabilize variance and improve normality. Additionally, scaling techniques such as min-max scaling (Han, Kamber, and Pei 2012) and z-score standardization (Aggarwal 2016) are frequently used to make data features easier to compare with each other. Min-max scaling adjusts features to a fixed range, while z-score standardization centers them around zero with a standard deviation of one (Hastie, Tibshirani, and Friedman 2009). The effectiveness of normality transformations can be evaluated using statistical tests such as the Shapiro-Wilk test, which measures how closely the data approximates a normal distribution (Shapiro and Wilk 1965). A test statistic closer to 1 and a p-value greater than 0.05 indicate that the data does not significantly deviate from normality (D. C. Montgomery, Jennings, and Kulahci 2015).

Building on this foundation, statistical methods are further divided into parametric and non-parametric approaches, depending on the data's distribution. Parametric methods, such as Z-Score and Gaussian Mixture Models (GMM), assume a known data distribution, typically a normal distribution, and rely on that assumption to identify outliers (Shumway and Stoffer 2000). Non-parametric methods, by contrast, make no assumptions about the data distribution (Wasserman 2006).

Among these, a widely used technique is the Interquartile Range (IQR) method, which identifies outliers by calculating the range between the first (Q1) and third quartiles (Q3) of the dataset, called the IQR. Outliers are defined as data points falling below $Q1 - 1.5 \times IQR$ or above $Q3 + 1.5 \times IQR$. However, the multiplier (1.5) can be adjusted based on the specific use case, with larger values used for lenient outlier detection to account for greater variability and smaller values for stricter thresholds to identify more outliers (Dallah and Sulieman 2024). The IQR method is particularly effective for small, skewed datasets due to its straightforward approach

and robustness to non-normality (John W. Tukey 1977).

Clustering-based methods detect outliers by identifying points that do not fit into any cluster or belong to small, sparse clusters (Han, Kamber, and Pei 2012). For instance, density-based clustering methods, like Density-Based Spatial Clustering of Applications with Noise (DBSCAN), classify points in dense regions as clusters and points in sparse regions as outliers (Deng 2020). These methods evaluate the density of points within a region without relying on assumptions about a specific distribution, making them effective in noisy or non-normally distributed datasets (Ester et al. 1996).

In contrast to clustering methods, which group points into well-defined clusters based on proximity or density, nearest-neighbors methods focus on analyzing the relationship between individual points and their immediate neighbors. These methods assume that "normal data instances occur in dense neighborhoods, while outliers are isolated from their nearest neighbors" (Hodge and Austin 2004). For instance, the Local Outlier Factor (LOF) is a widely used approach based on the assumption that the density surrounding normal data points is similar to that of their neighbors. In contrast, the density surrounding outliers is significantly lower, which distinguishes them from the denser clusters of normal data points (Tan, Steinbach, and Kumar 2006). By focusing on local density rather than global distance, LOF is robust to non-normal distributions (Hans-Peter Kriegel et al. 2009).

Machine learning-based methods for outlier detection have gained popularity since the mid-2000s, with approaches ranging from traditional algorithms to more complex deep learning techniques (Aggarwal 2016). Traditional machine learning techniques, such as Isolation Forest, identify outliers by recursively partitioning the data. At each iteration, the algorithm selects a feature at random and determines a split value within the feature's range. This process continues until individual data points are isolated. Outliers are detected as they tend to reside in sparse regions of the feature space, requiring fewer splits to isolate, whereas normal data points in denser regions necessitate more splits (Liu, Ting, and Zhou 2008). Isolation Forest is particularly well-suited for small, skewed datasets as it does not assume any underlying data distribution and efficiently detect outliers even when the data is unevenly distributed (Cheng, Zou, and Dong

2019).

2.1.3 Treatment of outliers

Once outliers have been identified in a dataset, it is crucial to manage them effectively to ensure accurate data analysis and modelling. The literature categorizes outlier treatment into three main strategies: removal, modification, and robust estimation.

The first approach, removal of outliers, is commonly used in situations where outliers are clearly erroneous or unrelated to the phenomena being studied. This method, often referred to as trimming, has been extensively discussed in the literature (Lix and Keselman 1998, Dixon and Yuen 1974). While trimming can improve the robustness of certain analyses, it carries the risk of discarding valuable data points, as noted by J. W. Tukey and Statistics 1959, who argued that indiscriminate removal of outliers may lead to the loss of significant information.

Winsorization offers an alternative by adjusting outlier values instead of removing them (Ghosh and Vogt 2012). In this method, outliers are replaced by more plausible values, such as the next highest or lowest value within a non-outlier range, or by a central tendency measure such as the mean, median, or mode (Liao, Li, and Brooks 2016). Winsorization reduces the influence of outliers while preserving the overall data structure. However, the choice of replacement values is critical for maintaining analytical integrity (Leys et al. 2019). For small and right-skewed datasets, median winsorization or quantile-based winsorization is particularly robust, as it limits the influence of extreme values while preserving the original skew in the data (Wilcox 2017).

The third approach involves robust estimation methods, which minimize the influence of outliers during the modeling process itself without explicitly identifying or modifying them. Unlike removal or modification techniques, robust estimation incorporates outliers into the analysis but down-weights their influence, allowing models to focus on the central structure of the data (Rousseeuw and Leroy 1987). Among the various robust estimation techniques, M-estimation is widely used for its ability to reduce the influence of extreme values by employing robust loss functions that assign smaller weights to observations with larger residuals (Hubert, Rousseeuw,

and Van Aelst 2008). For example, the Huber loss function applies squared loss to small residuals and switches to absolute loss for larger residuals, balancing sensitivity to central patterns and robustness to outliers (Huber 1964).

M-estimation, when combined with models such as SARIMA and XGBoost, introduces robustness directly into the modeling process by replacing traditional optimization methods with robust loss functions. In SARIMA, this adjustment occurs during parameter estimation, where smaller residuals contribute fully to the model, allowing it to capture temporal dependencies and seasonal patterns without distortion from outliers (Maronna et al. 2019). For XGBoost, the Huber loss function modifies the gradient boosting process by dynamically adjusting gradients to prioritize data points with smaller residuals while reducing the impact of outliers (Chen and Guestrin 2016). This approach preserves all data points, ensuring that both the target variable and features are handled robustly (Friedman 2001). By integrating M-estimation directly into the modeling process, SARIMA and XGBoost maintain their focus on the core data structure, enhancing model stability and predictive accuracy without requiring explicit outlier removal or replacement (Hampel et al. 2005).

2.1.4 Implications of treating outliers in the target variable and features

Outliers can occur in both the target variable and the modeling features, each with distinct implications due to their roles in the modeling process. Outliers in the target variable have a direct impact on the optimization process, as they influence error metrics used to evaluate model performance (Hampel et al. 2005). Extreme values can disproportionately affect metrics such as the mean squared error or mean absolute percentage error, leading to predictions skewed toward these observations (Chicco, Warrens, and Jurman 2021). Treatment of target outliers aims to stabilize model outputs and reduce sensitivity to these extreme values. However, this approach may inadvertently mask meaningful rare events, especially when those outliers represent significant phenomena rather than noise (John W. Tukey 1977).

In contrast, outliers in the modelling features primarily affect the relationships between predictors and the target variable, introducing distortions that can degrade model accuracy (Aggarwal 2016). Such distortions arise from the disproportionate influence of feature outliers on

the model's ability to accurately capture the relationship between input variables and the target. Distance- or density-based methods, such as clustering or nearest-neighbors algorithms, are particularly sensitive to such distortions due to their reliance on data point proximity (Ester et al. 1996). To mitigate the impact of feature outliers, preprocessing techniques are commonly employed. These methods aim to reduce the disproportionate influence of outliers on the modeling process while preserving the variability essential for accurate predictions (Hastie, Tibshirani, and Friedman 2009, Chen and Guestrin 2016). While addressing target outliers contributes to stable model outputs, managing feature outliers ensures the preservation of accurate predictor-target relationships.

The simultaneous presence of outliers in both the target variable and the features introduces significant complexities for model performance. Treating these outliers independently can result in inconsistencies due to the interdependence between these components (Maronna et al. 2019). On the other hand, excessive treatment of both target and feature outliers risks eliminating critical patterns that are essential for accurate predictions, particularly in small or complex datasets where variability is valuable (Rousseeuw and Leroy 1987).

2.2 Methodology

This chapter outlines the methodological approach used to analyze, detect, and treat outliers, as well as to optimize model performance. It begins with an analysis of the distribution of key variables and the application of normalization techniques to approximate normality where necessary. Next, the chapter details the procedures for outlier detection and treatment, using a combination of statistical, clustering and machine learning-based methods. Finally, it describes the process of hyperparameter tuning and model evaluation, where optimized models are compared with previously developed models.

2.2.1 Distribution analysis and normalization of variables

The initial step involved analyzing the distribution of the variables of interest to understand their behavior and determine whether transformation was necessary to achieve normality. This analysis was conducted to address the requirements of certain outlier detection techniques, as

discussed in section 2.1.2 (Unsupervised outliers' detection techniques), where data normality is emphasized as a condition. When the data did not follow a normal distribution, normalization techniques were applied to adjust the variables. Five methods were considered: log transformation, square root transformation, Box-Cox transformation, min-max scaling, and z-score standardization. These transformations were intended to align the variables more closely with the assumptions required by outlier detection techniques. The effectiveness of the transformations was evaluated using the Shapiro-Wilk test.

The target variable, i.e., the number of clients to flag, was the only variable used for prediction in the univariate models. Its distribution was examined to assess the impact of outliers and to understand its underlying characteristics. For multivariate modeling, as outlined in section 1.5.2 (Feature Engineering), the most critical features included call duration, hold time, total calls to the technical line, and client score. These variables, identified for their importance in capturing patterns in call behavior, were also subjected to distribution analysis.

2.2.2 Detection and treatment of outliers

The subsequent step involved identifying and addressing outliers in the variables. For this purpose, four detection techniques were applied: IQR, LOF, Isolation Forest, and DBSCAN, each representing a distinct approach to outlier detection. Outliers identified using these techniques were treated using two distinct approaches: removal and median winsorization. Additionally, M-estimation, incorporating the Huber loss function, was utilized as a robust alternative, incorporating outlier handling directly into the modeling process without the need for prior detection.

In the univariate analysis, nine distinct approaches were evaluated. These included the combination of the four outlier detection techniques with two treatment methods, removal and winsorization, applied directly to the target variable. M-estimation was evaluated as a separate approach as it incorporate outlier handling directly within the modeling process. For the multivariate modeling, a distinction was made between the target variable and the additional features due to their differing roles in the modeling process, as outlined in section 2.1.4 (Implications of treating outliers in the target variable and features). This distinction was addressed through two specific scenarios: one where the target variable was treated using the same outlier detection

and treatment methods as the features, and another where the target variable was left untreated, and only the features were addressed. This design enabled a detailed analysis of the impact of outlier treatment on both components and their respective contributions to model performance.

In total, eighteen approaches were examined. Sixteen of these resulted from pairing the four detection techniques with the two treatment methods across the two scenarios. The remaining two approaches involved the application of M-estimation, either to both the target variable and the features or exclusively to the features.

2.2.3 Hyperparameter tuning and evaluation

After treating the outliers, the grid search function described in section 1.5.3.3.6 (Tailored hyperparameter tuning) was applied to each combination. This process was used to identify the optimal model parameters based on the updated datasets. The identified parameters were then applied to execute the models, enabling a direct comparison of their performance with the univariate and multivariate models developed during the PBL phase.

The evaluation of model performance followed the approach outlined in section 1.5.3.3 (Model Evaluation Criteria), which emphasized a dual focus on accuracy and stability using the Mean Absolute Percentage Error (MAPE) and Stability Index (SI) as evaluation metrics. MAPE provided a relative measure of prediction error, aligning with NOS' operational requirements, while SI quantified day-to-day variability in predictions to ensure consistent workloads for the Detractor Squad. A weighting scheme was applied, assigning 60% importance to MAPE and 40% to SI, reflecting a prioritization of forecast accuracy while accounting for NOS' emphasis on stability in predictions.

To ensure meaningful comparisons, model predictions were evaluated against the operational call volume target using temporal cross-validation. This approach tested predictions on unseen data while preserving the chronological order of events, ensuring robust and realistic assessments of model performance.

2.3 Results

This section presents the findings from the distribution analysis and examines the impact of outlier handling strategies on the performance of univariate SARIMA models and multivariate XGBoost models. The analysis focused on the influence of outlier detection and treatment on predictive accuracy and stability, with comparisons to models developed during the PBL phase. Specific outlier treatment methods were evaluated for their effectiveness in improving model performance and achieving optimal trade-offs between accuracy and stability.

2.3.1 Distribution analysis

The distribution analysis revealed significant deviations from normality across the variables examined. Except for the score variable, none of the other variables achieved normality, even after the application of transformation techniques. For the target variable, the Box-Cox transformation produced the highest Shapiro-Wilk test statistic of 0.9733, which was closest to the desired value of 1. However, the corresponding p-value of 0.0166 remained below the 0.05 threshold, confirming that normality was not achieved. Similar patterns were observed for call duration, hold time and number of calls. Although the Box-Cox transformation yielded the highest test statistics for these variables, the p-values remained close to zero, failing to meet normality criteria. A detailed summary of the results for all transformations is provided in table A1 in the appendix.

The score variable, on the other hand, naturally exhibited a distribution closer to normal, with a Shapiro-Wilk test statistic of 0.9575 and a p-value of 0.266, well above the 0.05 threshold. This indicated that the score variable aligned closely with a normal distribution and did not require transformation.

Given these findings, and the inability of transformation techniques to establish normality for most variables, no transformations were applied to the data prior to conducting the outlier detection analysis. While it is possible that transformations could have improved model predictions even without achieving strict normality, this hypothesis was not tested due to time constraints.

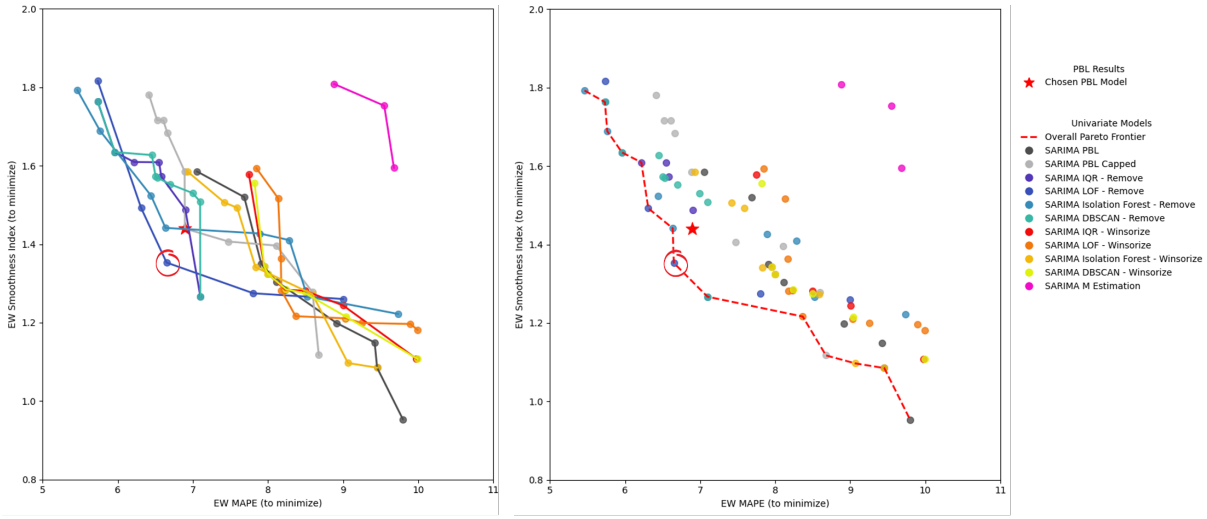
2.3.2 Univariate analysis

This section evaluates the performance of the models by comparing them to the univariate model chosen during the PBL phase, specifically the SARIMA PBL Capped $[(2,0,1),(0,0,1,20)]$ with a 20-day embed size presented in section 1.5.4.1 (Deployable Solution). The evaluation was guided by Pareto frontiers, constructed to identify configurations that achieved the optimal trade-off between predictive accuracy, measured by MAPE, and stability, assessed by SI. Individual Pareto frontiers were created for each model by plotting their performance across various parameter configurations. The Overall Pareto Frontier (OPF) was then established by comparing the best-performing configurations from each model's frontier, iterating to identify the points that achieved the most favorable balance between predictive accuracy and stability. The detailed methodology for creating the Pareto frontiers was outlined in section 1.5.4 (Results from PBL).

As shown in figure 7, models developed during the PBL phase, SARIMA PBL and SARIMA PBL Capped, clustered in the mid-range of the graph, reflecting moderate trade-offs between accuracy and stability. The chosen SARIMA PBL Capped model, marked by a red star, served as the reference point for assessing the performance of the new models. While some models demonstrated improvements in both metrics, others failed to surpass the performance of the chosen PBL model.

Several configurations achieved superior MAPE values while maintaining or improving stability. For instance, the SARIMA LOF - Remove model delivered better performance on both metrics, achieving a lower MAPE and enhanced stability compared to the PBL model. Similarly, SARIMA Isolation Forest - Remove improved MAPE while maintaining stability at levels comparable to the PBL models. Conversely, models such as SARIMA Isolation Forest - Winsorize achieved stability improvements without sacrificing accuracy, matching the PBL model's MAPE value while offering smoother predictions. By contrast, SARIMA M-Estimation resulted in higher values for both MAPE and SI, failing to improve the PBL model's performance.

A general trend emerged from the analysis: outlier removal techniques tended to improve



(a) Individual Pareto Frontiers (b) Overall Pareto Frontier

Figure 7: Pareto frontiers Univariate Models

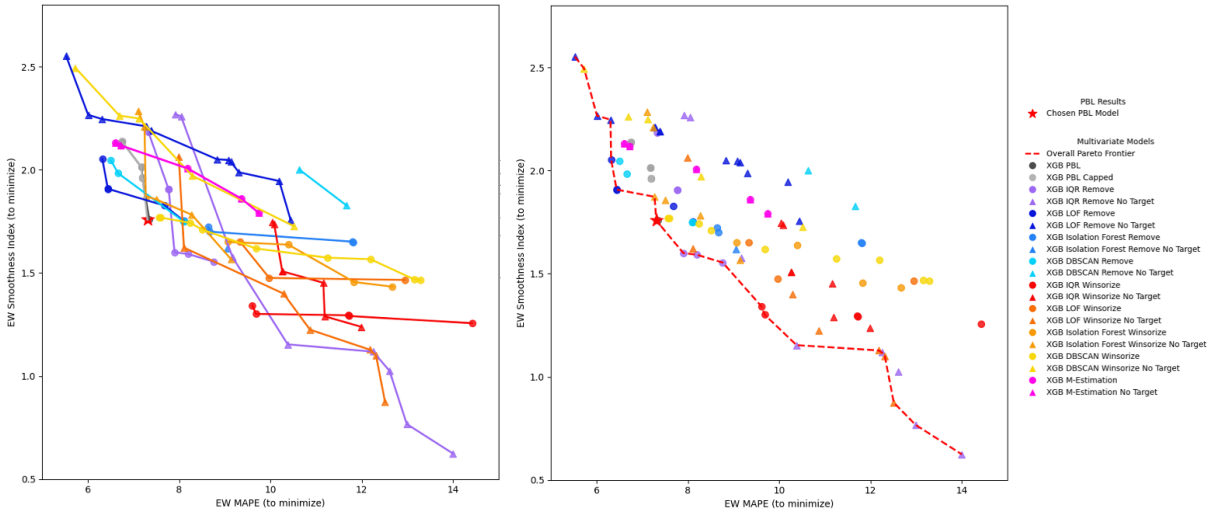
MAPE, indicating enhanced predictive accuracy, but often at the expense of higher SI, reflecting increased variability. Winsorization methods, particularly those paired with LOF or Isolation Forest detection, provided a more balanced trade-off by improving stability while maintaining acceptable MAPE values.

A closer examination of specific models revealed two configurations from the analysis as particularly noteworthy when compared to the chosen PBL model, which achieved a MAPE of 6.89 and an SI of 1.44. The SARIMA Isolation Forest - Remove configuration $[(1,0,2),(0,0,1,20)]$, with an embed size of 30, matched the PBL model's stability while improving MAPE by 3.77% (a reduction of 0.26 percentage points), bringing it down to 6.63. The SARIMA LOF - Remove configuration $[(1,0,1),(2,0,1,5)]$, with an embed size of 30, achieved the best overall performance. This configuration, represented by the red circle in figure 7, improved MAPE by 3.48% (0.24 percentage points) and SI by 6.25% (0.09 percentage points), achieving values of 6.65 and 1.35, respectively. These results reflected a favorable balance between accuracy and stability, aligning with NOS' preference for stable predictions while assigning slightly greater importance to accuracy.

2.3.3 Multivariate analysis

This section assessed the performance of multivariate forecasting models by comparing them to the reference model, PBL XGB, which utilized 35 estimators, an absolute error regression

objective, a maximum depth of 25, an L1 regularization term of 0.1, and a 10-day embed size, as described in section 1.5.4.2 (Build-Onto Solution). As in the univariate analysis, Pareto frontiers were employed to evaluate and visualize the trade-offs between MAPE and SI for each model, as illustrated in figure 8. A key distinction between combinations labeled as ”No Target” and those without this label is central to interpret the graph. As detailed in section 2.2.2 (Detection and treatment of outliers), a distinction was made between treatment of the target variable and features. In combinations labeled “No Target,” only the features were treated for outliers, while the target variable remained unchanged. In contrast, combinations without this label applied the same outlier detection and treatment method to both the target variable and the features.



(a) Individual Pareto Frontiers (b) Overall Pareto Frontier

Figure 8: Pareto frontiers Multivariate Models

A notable trend emerged from figure A1 in the appendix. Models labeled as ”No Target”, where only the features were treated for outliers, exhibited more extreme behaviors compared to models where both the target variable and features were treated. Notably, the ”No Target” models achieved the best MAPE values, reflecting the highest predictive accuracy, but this was accompanied by the worst SI values, indicating reduced stability. Conversely, some configurations also achieved the best SI values, demonstrating the highest stability, but at the expense of the worst MAPE values, reflecting diminished predictive accuracy. In contrast, configurations where both the target variable and the features were treated displayed a more balanced pattern, clustering near the center of the graph. These configurations offered a more favorable trade-off between accuracy and stability, avoiding the extremes of either exceptionally low MAPE with

high instability or high stability with poor accuracy.

The selected XGB PBL model, represented by the red star in figure 8, served as the baseline for evaluating the other models. None of the tested configurations outperformed the PBL solution in both MAPE and SI simultaneously. While models such as XGB LOF - Remove and XGB DBSCAN - Remove improved accuracy, these gains came at the expense of reduced smoothness, as indicated by higher SI values.

In contrast, most of the models demonstrated significant improvements in stability (SI) compared to the PBL model. Configurations such as XGB M-estimation and XGB Isolation Forest - Winsorization showed clear advantages in reducing instability, although they involved some trade-offs in accuracy. Additionally, XGB IQR - Remove and XGB IQR - Winsorize proved effective in enhancing model stability, achieving substantial gains in smoothness while maintaining a reasonable balance with accuracy. These configurations were located along the Overall Pareto Frontier, indicating that further improvements in SI could only be achieved by compromising some level of accuracy.

2.4 Discussion of the results

The results of the univariate analysis aligned with the findings from the PBL phase, where the SARIMA PBL Capped model outperformed the original SARIMA PBL model due to the incorporation of outlier detection and treatment. Building on these insights, the hypothesis that advanced outlier treatment improves both predictive accuracy (lower MAPE) and stability (lower SI) was confirmed. Models utilizing proper outlier detection and treatment methods demonstrated notable improvements in performance.

Among the outlier detection techniques, LOF emerged as the most effective for univariate modeling. It achieved the lowest MAPE value by accurately identifying and removing local outliers without distorting the dataset's overall structure. Its localized approach to detecting deviations allowed it to adapt to datasets with varying densities, making it particularly effective for datasets with non-linear or context-dependent variations. As a result, LOF preserved meaningful data patterns while reducing distortions, leading to strong predictive accuracy and moderate stability.

Isolation Forest also performed well but did not surpass LOF. Its aggressive approach to detecting outliers identified a larger number of outliers but introduced higher instability (SI) due to the model's increased sensitivity to extreme values. In comparison, the IQR method prioritized stability (lower SI) but lacked sensitivity to subtle outliers, resulting in higher MAPE. This conservative method primarily identified extreme values beyond the interquartile range, restricting the model's ability to capture nuanced variations. Similarly, DBSCAN exhibited lower sensitivity, detecting fewer outliers and focusing primarily on points distant from dense regions. While this method achieved low SI, its inability to capture critical outliers led to diminished predictive accuracy (higher MAPE).

The comparison of winsorization and removal techniques revealed a clear trade-off between stability and accuracy. Winsorization reduced instability by capping extreme values, resulting in smoother and more consistent predictions (lower SI). However, it introduced bias by forcing outliers closer to the majority, which reduced accuracy (higher MAPE). In contrast, removal techniques improved accuracy by eliminating extreme values, indicating that these outliers likely represented observations outside the population and did not provide valuable information for the model. Nevertheless, this increased sensitivity to the remaining data resulted in greater instability (higher SI).

The multivariate analysis provided several key insights into the performance of outlier detection techniques and their impact on model accuracy and stability. In comparison to the baseline model, PBL XGB, none of the tested configurations achieved simultaneous improvements in both MAPE and SI. This outcome highlighted the inherent difficulty in achieving a perfect balance between predictive accuracy and stability in multivariate forecasting. These findings aligned with results from the PBL phase, where the model's performance before and after capping remained consistent.

The "No Target" configurations exhibited more extreme behaviors during the analysis, achieving either very low MAPE (high accuracy) or very low SI (high stability). This could be attributed to the isolated treatment of features, leaving the target variable untreated. When only the features were treated, the cleaned feature set often dominated predictions, leading to higher

accuracy but increased sensitivity to noise in the uncleaned target variable. Conversely, when the target's influence diminished, models gained stability but sacrificed predictive accuracy. In contrast, configurations treating both the target variable and features displayed a more balanced trade-off between accuracy and stability. This uniform treatment likely reduced inconsistencies between the input and output distributions, enabling the models to generalize more effectively and avoid overfitting to noise in either the target or the features.

The consistent inability to improve both MAPE and SI simultaneously indicated the trade-off between accuracy and stability. Models prioritizing accuracy, such as those employing outlier removal, enhanced predictive performance by effectively eliminating true outliers. However, this reduction in data points introduced greater instability, as fewer observations remained for the model to generalize effectively. Conversely, models prioritizing stability employed more extensive data sanitization techniques, which smoothed the dataset and reduced variability. While this approach improved stability, it often came at the expense of predictive accuracy by erasing fine-grained patterns in the target variable.

Specific models, such as XGB LOF - Remove and XGB DBSCAN - Winsorize, demonstrated improvements in accuracy by identifying and treating outliers in a more restrained manner compared to other methods, detecting far fewer outliers than IQR or Isolation Forest. These methods focused on the most significant outliers, preserving the overall data distribution and achieving better MAPE results. However, this selective approach led to higher SI values, as fewer extreme data points were addressed, leaving residual instability in the model's predictions.

In contrast, methods such as XGB Isolation Forest - Winsorize and XGB IQR - Winsorize reduced instability (lower SI) by employing more aggressive outlier detection and treatment strategies. These configurations achieved smoother predictions but at the expense of accuracy, as indicated by higher MAPE values. The replacement of extreme values reduced variability but also eliminated data points that could have contributed to predictive accuracy.

Overall, most models outperformed the multivariate PBL baseline in terms of stability (SI). This improvement was likely due to more effective outlier treatment across both the target variable and features, which minimized the impact of extreme values and resulted in smoother, more

stable predictions.

2.5 Limitations & Future Work

This study offered valuable insights into the impact of outlier detection and treatment on forecasting model optimization but was subject to several limitations. Due to time constraints, the analysis focused on a limited set of outlier detection techniques, which, while effective for the dataset used, may not generalize well to datasets with different characteristics. Furthermore, the study did not explore the potential benefits of combining multiple outlier detection methods or employing advanced hybrid techniques, which could improve model performance.

A significant constraint was the small size of the dataset. While based on real-world time series data, the limited sample size restricted the generalizability of the findings to datasets with more complex structures or higher-dimensional features. This limitation also constrained the testing of more advanced models, such as neural networks, which typically require larger datasets to fully realize their potential. As a result, the conclusions may not extend to larger, more complex datasets.

Finally, the study concentrated on predictive accuracy and stability, evaluated through MAPE and SI, without addressing other aspects of model performance, such as interpretability, scalability, and computational efficiency. Future research could address these gaps by exploring automated pipelines, hybrid outlier detection methods, and advanced machine learning approaches tailored for dynamic and large-scale datasets.

2.6 Conclusion

The results of this study highlight the critical role of outlier detection and treatment in improving forecasting models, with a particular focus on balancing predictive accuracy and stability. Both univariate and multivariate analyses demonstrated the trade-off between these metrics, where enhancing one often led to a decline in the other. In the univariate analysis, SARIMA models incorporating techniques such as LOF and Isolation Forest showed notable improvements in performance. The SARIMA LOF - Remove configuration, in particular, demonstrated the

effectiveness of LOF's local density approach, which preserved essential data patterns while effectively eliminating outliers. This configuration achieved improvements in predictive accuracy and stability, underscoring the value of localized outlier detection methods.

In the multivariate analysis, none of the tested configurations surpassed the baseline PBL model by simultaneously improving both MAPE and SI. While the XGB LOF - Remove configuration improved accuracy, the XGB IQR - Remove configuration achieved greater stability while maintaining reasonable accuracy. Models treating both the target variable and the features achieved a more balanced trade-off, resulting in smoother and more consistent predictions. In contrast, configurations focusing solely on feature treatment exhibited extreme behaviors, achieving either high accuracy with reduced stability or the reverse.

In conclusion, this study demonstrated that outlier detection and treatment are crucial for optimizing forecasting models, particularly for datasets with significant variability. The LOF - Remove configuration emerged as a robust approach in both univariate and multivariate contexts, offering valuable insights for model improvement.

References

Aggarwal, Charu C. (Nov. 2016). *Outlier Analysis*. Cham, Switzerland: Springer International Publishing. DOI: 10.1007/978-3-319-47578-3.

Allende, Héctor and Carlos Valle (Oct. 2016). "Ensemble Methods for Time Series Forecasting". In: *Claudio Moraga: A Passion for Multi-Valued Logic and Soft Computing*. Ed. by Rudolf Seising and Héctor Allende-Cid. Springer International Publishing, pp. 217–232. ISBN: 978-3-319-48317-7. DOI: 10.1007/978-3-319-48317-7_13.

Assimakopoulos, Vassilis and Konstantinos Nikolopoulos (Dec. 2000). "The theta model: a decomposition approach to forecasting". In: *International Journal of Forecasting* 16.4, pp. 521–530. DOI: 10.1016/S0169-2070(00)00066-2.

Barnett, V. and T. Lewis (1984). *Outliers in Statistical Data*. Wiley Series in Probability and Mathematical Statistics. Chichester: Wiley. ISBN: 9780471995999. URL: https://books.google.pt/books?id=AhmoAAAAIAAJ.

Bastianin, Andrea, Marzio Galeotti, and Matteo Manera (June 2019). "Statistical and economic evaluation of time series models for forecasting arrivals at call centers". In: *Empirical Economics* 57.3, pp. 923–955. DOI: 10.1007/s00181-018-1475-y.

Bergmeir, Christoph and José M. Benítez (2012). "On the Use of Cross-Validation for Time Series Predictor Evaluation". In: *Information Sciences* 191, pp. 192–213. DOI: 10.1016/j.ins.2011.12.028.

Berrar, Daniel et al. (2019). "Cross-validation". In: *Encyclopedia of Bioinformatics and Computational Biology* 1, pp. 542–545.

Box, George EP et al. (Mar. 2016). “Time series analysis: forecasting and control”. In: *Journal of Time Series Analysis*. Ed. by Granville Tunnicliffe Wilson. Vol. 37. 5. John Wiley & Sons, pp. 712. DOI: 10.1111/jtsa.12194.

Brown, R.G. (Jan. 1959). *Statistical Forecasting for Inventory Control*. Vol. 2. New York: McGraw-Hill, pp. 443–473. URL: https://books.google.es/books?id=oKI8AAAAIAAJ.

Bühlmann, Peter (2012). “Bagging, Boosting, and Ensemble Methods”. In: *Handbook of Computational Statistics: Concepts and Methods*, pp. 985–1022. DOI: 10.1007/978-3-642-21551-3_31.

Cerqueira, Vitor, Luis Torgo, and Igor Mozetič (2020). “Evaluating Time Series Forecasting Models: An Empirical Study on Performance Estimation Methods”. In: *Machine Learning* 109.11, pp. 1997–2028. DOI: 10.1007/s10994-020-05910-7.

Chakraborty, Kanad et al. (Nov. 1992). “Forecasting the behavior of multivariate time series using neural networks”. In: *Neural Networks* 5.6, pp. 961–970. DOI: 10.1016/s0893-6080(05)80092-9.

Chatfield, Chris (Oct. 2000). *Time-series forecasting*. 1. Boca Raton: Chapman and Hall/CRC. DOI: 10.1201/9781420036206.

Chen, Tianqi and Carlos Guestrin (Mar. 2016). “Xgboost: A scalable tree boosting system”. In: *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794.

Cheng, Zhangyu, Chengming Zou, and Jianwei Dong (Oct. 2019). “Outlier Detection Using Isolation Forest and Local Outlier Factor”. In: *Proceedings of the Conference on Research in Adaptive and Convergent Systems*. ACM, pp. 161–168. DOI: 10.1145/3338840.3355641.

Chicco, Davide, Matthijs J. Warrens, and Giuseppe Jurman (July 2021). "The Coefficient of Determination R-squared Is More Informative Than SMAPE, MAE, MAPE, MSE, and RMSE in Regression Analysis Evaluation". In: *PeerJ Computer Science* 7, e623. DOI: 10.7717/peerj-cs.623.

Connor, J.T., R.D. Martin, and L.E. Atlas (Mar. 1994). "Recurrent neural networks and robust time series prediction". In: *IEEE Transactions on Neural Networks* 5.2, pp. 240–254. DOI: 10.1109/72.279188.

Dallah, Dania and Hana Sulieman (Jan. 2024). *Outlier Detection Using the Range Distribution*. Cham: Springer International Publishing, pp. 687–697. DOI: 10.1007/978-3-031-41420-6\{_}57.

Davey, AM and BE Flores (Sept. 1993). "Identification of seasonality in time series: A note". In: *Mathematical and Computer Modelling* 18.6, pp. 73–81. DOI: 10.1016/0895-7177(93)90126-J.

De Gooijer, Jan G. and Rob J. Hyndman (Jan. 2006). "25 years of time series forecasting". In: *International Journal of Forecasting* 22.3, pp. 443–473. DOI: 10.1016/j.ijforecast.2006.01.001.

Deng, Dingsheng (Sept. 2020). "DBSCAN Clustering Algorithm Based on Density". In: *Proceedings of the 7th International Forum on Electrical Engineering and Automation (IFEEA)*. IEEE, pp. 949–953. DOI: 10.1109/ifeea51475.2020.00199.

Dietterich, Thomas G (Jan. 2000). "Ensemble Methods in Machine Learning". In: *Multiple Classifier Systems*. Springer Berlin Heidelberg, pp. 1–15. DOI: 10.1007/3-540-45014-9_1.

Dixon, Wilfrid J. and Kareb K. Yuen (1974). "Trimming and Winsorization: A Review". In: *Statistische Hefte* 15.2, pp. 157–170.

Duan, Lian et al. (June 2008). “Cluster-Based Outlier Detection”. In: *Annals of Operations Research* 168.1, pp. 151–168. DOI: 10.1007/s10479-008-0371-9.

Ehrgott, Matthias (2012). “Vilfredo Pareto and multi-objective optimization”. In: *Doc. math* 8, pp. 447–453.

Ester, Martin et al. (1996). “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise”. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96)*. Vol. 96. 34. AAAI Press, pp. 226–231. ISBN: 1-57735-004-9. URL: https://www2.cs.uh.edu/~ceick/7363/Papers/dbscan.pdf.

Falessi, Davide et al. (2020). “On the Need of Preserving Order of Data When Validating Within-Project Defect Classifiers”. In: *Empirical Software Engineering* 25, pp. 4805–4830. DOI: 10.1007/s10664-020-09858-3.

Friedman, Jerome H. (Oct. 2001). “Greedy Function Approximation: A Gradient Boosting Machine”. In: *The Annals of Statistics* 29.5, pp. 1189–1232. DOI: 10.1214/aos/1013203451.

Ghosh, Dhiren and Andrew Vogt (Aug. 2012). “Outliers: An Evaluation of Methodologies”. In: *Proceedings of the Joint Statistical Meetings*. Vol. 12. 1. American Statistical Association, pp. 3455–3460. URL: https://www.asasrms.org/Proceedings/y2012/Files/304068_72402.pdf.

Goodfellow, Ian, Yoshua Bengio, and Aaron Courville (Nov. 2016). *Deep Learning*. Cambridge: MIT Press, p. 775. DOI: 10.5555/3086952.

Grubbs, Frank E. (Feb. 1969). “Procedures for Detecting Outlying Observations in Samples”. In: *Technometrics* 11.1, pp. 1–21. DOI: 10.1080/00401706.1969.10490657.

Hamilton, James D. (Dec. 1994). *Time series analysis*. Princeton: De Gruyter and Jstor. DOI: 10.1515/9780691218632.

Hampel, Frank R. et al. (Mar. 2005). *Robust Statistics*. Hoboken: John Wiley & Sons. DOI: 10.1002/9781118186435.

Han, Jiawei, Micheline Kamber, and Jian Pei (June 2012). "Data Mining: Concepts and Techniques". In: *Techniques, Waltham: Morgan Kaufmann Publishers*. DOI: 10.1016/C2009-0-61819-5.

Hastie, Trevor, Robert Tibshirani, and Jerome H. Friedman (Feb. 2009). *The Elements of Statistical Learning*. New York: Springer New York. DOI: 10.1007/978-0-387-84858-7.

Hawkins, D. M. (Jan. 1980). *Identification of Outliers*. Dordrecht, Netherlands: Springer. DOI: 10.1007/978-94-015-3994-4.

Healy, M.J.R. and R.G. Brown (Jan. 1964). "Smoothing, Forecasting and Prediction of Discrete Time Series." In: *Journal of the Royal Statistical Society Series A (General)* 127.2, p. 292. DOI: 10.2307/2344012.

Hinton, Geoffrey and Terrence J. Sejnowski (1999). *Unsupervised Learning: Foundations of Neural Computation*. Cambridge: MIT Press. DOI: 10.7551/mitpress/7011.001.0001.

Hjorth, J. S. Urban (1994). *Computer intensive statistical methods. Validation, model selection, and bootstrap*. London: Chapman and Hall/CRC. DOI: 10.1201/9781315140056.

Hochreiter, Sepp and Jürgen Schmidhuber (Nov. 1997). "Long Short-term Memory". In: *Neural Computation* 9.8, pp. 1735–1780. DOI: 10.1162/neco.1997.9.8.1735.

Hodge, Victoria J. and Jim Austin (Oct. 2004). "A Survey of Outlier Detection Methodologies". In: *Artificial Intelligence Review* 22.2, pp. 85–126. DOI: 10.1007/s10462-004-4304-y.

Holt, Charles C. (Jan. 2004). "Forecasting seasonals and trends by exponentially weighted moving averages". In: *International Journal of Forecasting* 20.1, pp. 5–10. DOI: 10.1016/j.ijforecast.2003.09.015.

Huber, Peter J. (Mar. 1964). "Robust Estimation of a Location Parameter". In: *The Annals of Mathematical Statistics* 35.1, pp. 73–101. DOI: 10.1214/aoms/1177703732.

Hubert, Mia, Peter J. Rousseeuw, and Stefan Van Aelst (Feb. 2008). "High-Breakdown Robust Multivariate Methods". In: *Statistical Science* 23.1, pp. 92–119. DOI: 10.1214/088342307000000087.

Hyndman, Rob and George Athanasopoulos (July 2012). *Principles of Business Forecasting by Keith Ord & Robert Fildes Forecasting: Principles and Practice*.

Hyndman, Rob J. and George Athanasopoulos (2018a). *Forecasting: Principles and Practice*. 2nd. Melbourne: OTexts. DOI: 10.5281/zenodo.3740164.

– (May 2018b). *Forecasting: Principles and Practice*. Melbourne, Australia: OTexts. URL: https://otexts.com/fpp2/.

Koopman, Siem Jan and Kai Ming Lee (Mar. 2009). "Seasonality with Trend and Cycle Interactions in Unobserved Components Models". In: *Journal of the Royal Statistical Society Series C: Applied Statistics* 58.4, pp. 427–448. DOI: 10.1111/j.1467-9876.2009.00661.x.

Korstanje, Joos (July 2021a). *Advanced Forecasting with Python*. New York: Apress.

– (July 2021b). *Advanced forecasting with Python. With State-of-the-Art-Models Including LSTMs, Facebook's Prophet, and Amazon's DeepAR*. 1. Berkeley: Apress Berkeley, CA. DOI: 10.1007/978-1-4842-7150-6.

Kriegel, Hans-Peter et al. (Nov. 2009). "LoOP: Local Outlier Probabilities". In: *Proceedings of the 18th ACM Conference on Information and Knowledge Management*. Association for Computing Machinery, pp. 1649–1652. DOI: 10.1145/1645953.1646195.

LeCun, Yann, Yoshua Bengio, and Geoffrey Hinton (2015). “Deep Learning”. In: *Nature* 521.7553, pp. 436–444. DOI: 10.1038/nature14539.

Leys, Christophe et al. (Jan. 2019). “How to Classify, Detect, and Manage Univariate and Multivariate Outliers, with Emphasis on Pre-Registration”. In: *International Review of Social Psychology* 32.1. DOI: 10.5334/irsp.289.

Liao, Hongjing, Yanju Li, and Gordon Brooks (May 2016). “Outlier Impact and Accommodation Methods: Multiple Comparisons of Type I Error Rates”. In: *Journal of Modern Applied Statistical Methods* 15.1, pp. 452–471. DOI: 10.22237/jmasm/1462076520.

Lim, Bryan and Stefan Zohren (Feb. 2021). “Time-series forecasting with deep learning: a survey”. In: *Philosophical Transactions of the Royal Society A* 379.2194. DOI: 10.1098/rsta.2020.0209.

Liu, Fei Tony, Kai Ming Ting, and Zhi-Hua Zhou (Dec. 2008). “Isolation Forest”. In: *Proceedings of the 2008 Eighth IEEE International Conference on Data Mining*. IEEE, pp. 413–422. DOI: 10.1109/ICDM.2008.17.

Lix, Lisa M. and H. J. Keselman (June 1998). “To Trim or Not to Trim: Tests of Location Equality Under Heteroscedasticity and Nonnormality”. In: *Educational and Psychological Measurement* 58.3, pp. 409–429. DOI: 10.1177/0013164498058003004.

Maronna, Ricardo A. et al. (Jan. 2019). *Robust Statistics: Theory and Methods (with R)*. Hoboken: John Wiley & Sons. DOI: 10.1002/9781119214656.

Montgomery, D.C. and G.C. Runger (2010). *Applied Statistics and Probability for Engineers*. Hoboken: John Wiley & Sons. URL: https://books.google.es/books?id=_f4KrEcNAfEC.

Montgomery, Douglas C., Cheryl L. Jennings, and Murat Kulahci (Apr. 2015). *Introduction to Time Series Analysis and Forecasting*. Hoboken: John Wiley & Sons. URL: https://

handoutset.com/wp-content/uploads/2022/07/Introduction-to-Time-Series-Analysis-and-Forecasting-Douglas-C.-Montgomery-Cheryl-L.-Jennings-etc.-.pdf.

Muñoz-Garcia, J., J. L. Moreno-Rebollo, and A. Pascual-Acosta (Dec. 1990). "Outliers: A Formal Approach". In: *International Statistical Review/Revue Internationale de Statistique* 58.3, pp. 215–226. DOI: 10.2307/1403805.

Papadimitriou, S. et al. (2003). "LOCI: Fast Outlier Detection Using the Local Correlation Integral". In: *Proceedings of the 19th International Conference on Data Engineering (ICDE)*. IEEE, pp. 315–326. DOI: 10.1109/ICDE.2003.1260802.

Petropoulos, Fotios et al. (Sept. 2022). "Forecasting: theory and practice". In: *International Journal of Forecasting* 38.3, pp. 705–871. DOI: 10.1016/j.ijforecast.2021.11.001.

Pindyck, Robert S et al. (Feb. 2018). *Microeconomics*. 9. Upper Saddle River: Pearson. ISBN: 978-0134184241.

Płaza, Mirosław and Łukasz Pawlik (Mar. 2021). "Influence of the contact center systems development on key performance indicators". In: *IEEE Access* 9, pp. 44580–44591. DOI: 10.1109/ACCESS.2021.3066801.

Purba, Johana Sihol Marito and Juliza Hidayati (2020). "Performance improvement strategies to increase call center service level: a literature review". In: *IOP Conference Series: Materials Science and Engineering* 801.1, p. 012147. DOI: 10.1088/1757-899X/801/1/012147.

Rousseeuw, Peter J. and Mia Hubert (Nov. 2017). "Anomaly Detection by Robust Statistics". In: *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery* 8.2. DOI: 10.1002/widm.1236.

Rousseeuw, Peter J. and Annick M. Leroy (Oct. 1987). *Robust Regression and Outlier Detection*. New York: John Wiley & Sons. DOI: 10.1002/0471725382.

Shapiro, Samuel Sanford and Martin B. Wilk (Dec. 1965). "An Analysis of Variance Test for Normality (Complete Samples)". In: *Biometrika* 52.3-4, pp. 591–611. DOI: 10.1093/biomet/52.3-4.591.

Shmueli, Galit and Julia Polak (Feb. 2024). *Practical Time Series Forecasting with R: A Hands-On Guide*. Third. New York: Axelrod Schnall Publishers, p. 252. ISBN: 9780997847949. URL: https://www.barnesandnoble.com/w/practical-time-series-forecasting-with-r-julia-polak/1144996927 (visited on 12/09/2024).

Shumway, Robert H. and David S. Stoffer (2000). *Time Series Analysis and Its Applications*. Vol. 3. New York: Springer. DOI: 10.1007/978-1-4757-3264-5.

– (2017). "ARIMA Models". In: *Time Series Analysis and Its Applications: With R Examples*, pp. 75–163. DOI: 10.1007/978-3-319-52452-8.

Tan, P. N., M. Steinbach, and V. Kumar (May 2006). *Introduction to Data Mining*. Pearson International Edition. Boston: Pearson Addison Wesley. URL: https://books.google.es/books?id=_XdrQgAACAAJ.

Taylor, James W. (Dec. 2007). "A Comparison of Univariate Time Series Methods for Forecasting Intraday Arrivals at a Call Center". In: *Management Science* 54.2, pp. 253–265. DOI: 10.1287/mnsc.1070.0786.

Taylor, Sean J and Benjamin Letham (Apr. 2018). "Forecasting at Scale". In: *The American Statistician* 72.1, pp. 37–45. DOI: 10.1080/00031305.2017.1380080.

Torres, José F. et al. (Feb. 2021). "Deep Learning for Time Series Forecasting: A Survey". In: *Big Data* 9.1, pp. 3–21. DOI: 10.1089/big.2020.0159.

Tukey, J. W. and Princeton University. Department of Statistics (1959). *A Survey of Sampling from Contaminated Distributions*. STRG Technical Report. Princeton: Princeton University. URL: https://books.google.es/books?id=leRbzwEACAAJ.

Tukey, John W. (Jan. 1977). *Exploratory data analysis*. Reading: Pearson. ISBN: 978-0201076165.

Vagropoulos, Stylianos I. et al. (2016). "Comparison of SARIMAX, SARIMA, Modified SARIMA and ANN-Based Models for Short-Term PV Generation Forecasting". In: *2016 IEEE International Energy Conference (ENERGYCON)*. IEEE, pp. 1–6. DOI: 10.1109/ENERGYCON.2016.7513987.

Vishwas, BV and Ashish Patel (Aug. 2020). *Hands-on Time Series Analysis with Python. From Basics to Bleeding Edge Techniques*. 1. Berkeley: Apress Berkeley, CA. DOI: 10.1007/978-1-4842-5992-4.

Wackerly, Dennis D., William Mendenhall, and Richard L. Scheaffer (2014). *Mathematical Statistics with Applications*. Boston: Cengage Learning. URL: https://books.google.be/books?id=lTgGAAAAQBAJ.

Wang, Feng and Joey Aviles (May 2023). "Contrasting Univariate and Multivariate Time Series Forecasting Methods for Sales: A Comparative Analysis". In: *Applied Science and Innovative Research* 7.2, p127. DOI: 10.22158/asir.v7n2p127.

Wang, Yan and Yuankai Guo (Mar. 2020). "Forecasting method of stock market volatility in time series data based on mixed model of ARIMA and XGBoost". In: *China Communications* 17.3, pp. 205–221.

Wasserman, Larry (Oct. 2006). *All of Nonparametric Statistics*. New York: Springer New York. DOI: 10.1007/0-387-30623-4.

Wilcox, Rand R. (Aug. 2017). *Modern Statistics for the Social and Behavioral Sciences: A Practical Introduction*. Boca Raton: Chapman and Hall/CRC. DOI: 10.1201/9781315154480.

Winters, Peter R. (Apr. 1960). "Forecasting sales by exponentially weighted moving averages". In: *Management science* 6.3, pp. 324–342. DOI: 10.1287/mnsc.6.3.324.

Wu, Wei Biao and Zhibiao Zhao (May 2007). "Inference of trends in time series". In: *Journal of the Royal Statistical Society Series B: Statistical Methodology* 69.3, pp. 391–410. DOI: 10.1111/j.1467-9868.2007.00594.x.

Yao, Hui et al. (Apr. 2023). "Advanced industrial informatics towards smart, safe and sustainable roads: A state of the art". In: *Journal of traffic and transportation engineering (English edition)* 10.2, pp. 143–158. DOI: 10.1016/j.jtte.2023.02.001.

Yildiz, Cem Yarkin and O Erhun Kundakcioglu (Feb. 2024). "Optimization for Time Series Decomposition". In: *Encyclopedia of Optimization*. Springer Nature, pp. 1–9. DOI: 10.1007/978-3-030-54621-2_881-1.

Yule, George Udny (Jan. 1927). "VII. On a method of investigating periodicities disturbed series, with special reference to Wolfer's sunspot numbers". In: *Philosophical Transactions of the Royal Society of London. Series A, Containing Papers of a Mathematical or Physical Character* 226.636-646, pp. 267–298. DOI: 10.1098/rsta.1927.0007.

Zhang, Kaimeng, Chi Tim Ng, and Myung Hwan Na (Apr. 2020). "Real time prediction of irregular periodic time series data". In: *Journal of Forecasting* 39.3, pp. 501–511. DOI: 10.1002/for.2637.

Zimek, Arthur, Erich Schubert, and Hans-Peter Kriegel (Aug. 2012). "A Survey on Unsupervised Outlier Detection in High-Dimensional Numerical Data". In: *Statistical Analysis and Data Mining: The ASA Data Science Journal* 5.5, pp. 363–387. DOI: 10.1002/sam.11161.

A Appendix - Enhancing the performance of time series models through Unsupervised Outlier Detection and Treatment Techniques

A.1 Shapiro-Wilk Test Results

Table A1: Shapiro-Wilk Test Results for Different Transformation Techniques on Selected Variables

Variable	Transformation Technique	Shapiro-Wilk Test Statistic	p-value
Target	Log Transformation	0.853	p <0.001
	Square Root Transformation	0.761	p <0.001
	Box-Cox Transformation	0.973	0.0166
	Min-Max Scaling	0.644	p <0.001
	Z-Score Standardization	0.644	p <0.001
Call Duration	Log Transformation	0.992	p <0.001
	Square Root Transformation	0.878	p <0.001
	Box-Cox Transformation	0.998	p <0.001
	Min-Max Scaling	0.480	p <0.001
	Z-Score Standardization	0.480	p <0.001
Holdtime	Log Transformation	0.891	p <0.001
	Square Root Transformation	0.636	p <0.001
	Box-Cox Transformation	0.928	p <0.001
	Min-Max Scaling	0.255	p <0.001
	Z-Score Standardization	0.255	p <0.001
Number of calls	Log Transformation	0.861	p <0.001
	Square Root Transformation	0.917	p <0.001
	Box-Cox Transformation	0.970	p <0.001
	Min-Max Scaling	0.949	p <0.001
	Z-Score Standardization	0.949	p <0.001

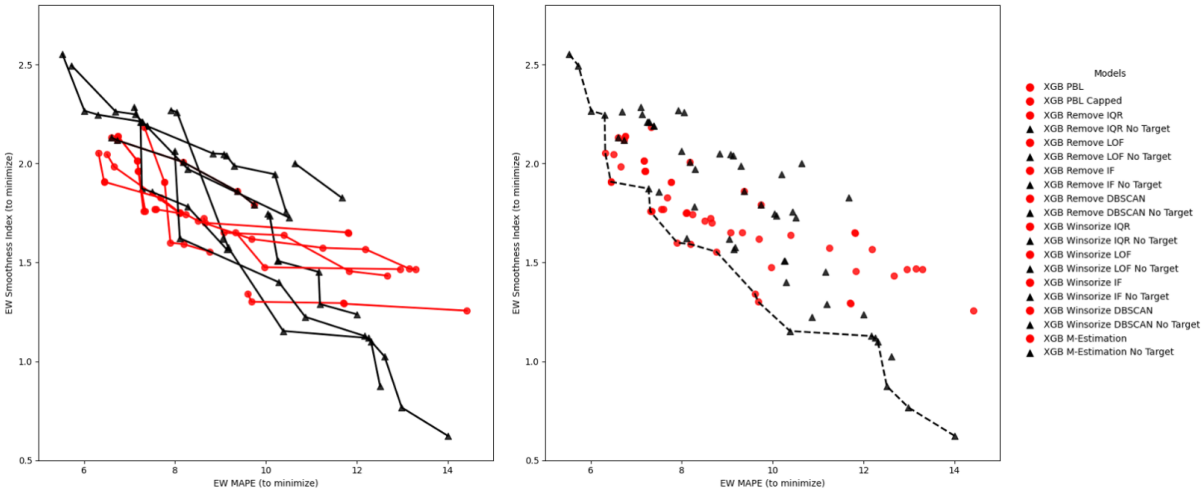
Table A1 presents the results of the Shapiro-Wilk test conducted to assess the normality of the selected variables (Target, Call Duration, Holdtime, and Number of Calls) after applying various transformation techniques. The Shapiro-Wilk test statistic and associated p-values are reported for each transformation method, including Log Transformation, Square Root Transformation, Box-Cox Transformation, Min-Max Scaling, and Z-Score Standardization.

The results indicate that none of the transformations succeeded in fully normalizing the vari-

ables, as evidenced by p-values below the 0.05 threshold.

Among the tested variables, the Call Duration and Number of Calls variables showed relatively higher Shapiro-Wilk test statistics for the Box-Cox transformation, indicating it was the most effective method across variables. In contrast, Min-Max Scaling and Z-Score Standardization consistently returned lower test statistics, suggesting limited efficacy in reducing skewness and approximating normality.

A.2 XGBoost Pareto Frontier



(a) Individual Pareto Frontiers (b) Overall Pareto Frontier

Figure A1: Pareto Frontiers for Multivariate Models: Comparison of Target-Treated and Feature-Only Models

Figure A1 presents the Pareto frontiers for multivariate XGBoost models, highlighting the trade-offs between predictive accuracy (MAPE) and stability (SI). Panel (a) displays the Individual Pareto Frontiers, where each configuration's performance across various parameter settings is shown. Models labeled as "No Target" (black markers) focus on treating outliers only in the features, while others (red markers) applied outlier treatment to both the target variable and features. A clear distinction emerges, with the "No Target" models achieving lower MAPE values (higher accuracy) but exhibiting significantly higher SI values (lower stability). These behaviors illustrate the extremes in performance when the target variable remains untreated.

In panel (b), the Overall Pareto Frontier consolidates the best-performing configurations across all individual frontiers. Models treating both the target variable and features (red) appear to cluster toward the center, representing a more balanced trade-off between accuracy and stability. This suggests that uniform outlier treatment across the dataset helps avoid extremes, delivering consistent and reliable predictive outcomes.