NOVA IMS Information Management School

MDSAA

Master Degree Program in

Data Science and Advanced Analytics

FORECASTING REVENUE: BUDGET SUPPORTING TOOL

Forecasting SGB Revenue

João Miguel Amaral Carvalho

Internship Report

presented as partial requirement for obtaining the Master Degree Program in Data Science and Advanced Analytics

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

Universidade Nova de Lisboa

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

FORECASTING REVENUE: BUDGET SUPPORTING TOOL

by

João Miguel Amaral Carvalho

Internship report presented as partial requirement for obtaining the Master's degree in Advanced Analytics, with a Specialization in Data Science.

Supervisor: Professor Doutor Flávio Luís Portas Pinheiro

11/2023

STATEMENT OF INTEGRITY

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

João Miguel Amaral Carvalho

Lisboa, 27 de Novembro de 2023

ACKNOWLEDGEMENTS

I want to thank all my friends and colleagues that have accompanied me during the master's as well as during the project. More specifically to my NOVA IMS friends, colleagues and professor: they have been walking side by side with me for 5 years and have always helped me passing through any academical difficulty; To my family and friends for all the continuous support; And to anyone which have supported me in some way. Also, a big thank you to all my Siemens colleagues which have always been very kind to me, and more specifically to all my team members who received me so well and made me feel like one of them since the beginning. To Mário Gonçalves for giving me the opportunity to work on this very big family which is Siemens. And lastly to my NOVA IMS mentor, Doctor Flávio Pinheiro which was also my professor during the master's degree and have always been so kind and helpful since the beginning and during the development of the internship.

ABSTRACT

Data-driven solutions paired with Artificial Intelligence and Machine Learning´s rapid expansion are increasingly becoming an interest topic for companies who want to keep up with the current trends. But there is a gap between the knowledge already established in the enterprise world and the enterprise realm and the burgeoning techniques that are continually emerging. The developed project aimed to fill that gap transferring knowledge from the new developers to the business environment while developing a real-world use-case for budgeting and performance control using time-series forecasting. Using python and recurring to methods like ARIMA and Prophet we could develop a successful application which can compete with the already existing procedures while also serve as a catalyst for generating interest in these evolving fields and set the pace for new upcoming use cases in the area.

KEYWORDS

Time-Series Forecasting; ARIMA; Prophet; Machine Learning; Data Science.

Sustainable Development Goals (SGD):

INDEX

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ACF Auto-correlation function

AR Auto-regressive

ARIMA Autoregressive Integrated Moving Average

AWS Amazon Web Services

BST Budget Supporting Tool

HANA High-Performance Analytic Appliance

KPI Key Performance Indicator

MA Moving Average

MAE Mean Absolute Error

MFC Manual Forecast

PACF Parcial auto-correlation function

Q-Q Quantile-Quantile (plot)

SARIMA Seasonal autoregressive Integrated Moving Average

SGB Siemens General Business

1. INTRODUCTION

Planning is something we all do, without planning is too easy to get lost or waste resources which in hindsight would have been better allocated elsewhere. Practices such as "Budgetary Control" that seek to devolve decision making to departments were established back in the 1920's when James McKinsey, who later became the founder of the McKinsey firm wrote a book of the same name. Organisations are taught that a key exercise is to set annual budgets to allocate resources and then track performance against them (Coveney & Cokins, 2017). Budgeting is an important part of the implementation of a company's strategy - it helps executives recognize and solidify annual directions of development and planning while Budget Controlling keep the company on track toward its financial goals. The actions of analysing budget differences comparing it against actuals and making strategic corrections have central roles in the performance control process (Passoja, 2015).

Today´s business landscape has reduced the ability to forecast accurately. This uncertainty over the future comes from advances in technology, in particular the Internet. In the past years, it has removed geographic boundaries and enabled an easier connection between suppliers and customers. Moreover, online social networks and communities such as Facebook, Instagram, Twitter can apply significant influence on the social trend and therefore on customers' buying behaviours (Coveney & Cokins, 2017). With the proliferation of digital technologies and the abundance of data by them generated, companies across all industries are realizing the potential of time-series forecasting to improve their operations, enhance customer experiences, and drive innovation. When applied to financial data, accurately forecasting sales and costs can play a very important role in budget management and maximizing profits of a company (Ensafi, Hassanzadeh Amin, Zhang, & Shah, 2022) and so it is increasingly becoming a key topic in the enterprise world.

The BST – Budget Supporting Tool is a Siemens tool which resources to time-series analysis with the final purpose of fulfilling the need for an extra and reliable resource when forecasting certain KPI's along different levels in the Siemens´ department SGB – Siemens General Business. More specifically, diving into the financial department of the SGB, the BST is owned by the Finance Controlling sector.

We will dive on how forecasting techniques can impact the trail to success of a company and how they can be implemented. More specifically, focusing on the following topics:

- To examine the importance of forecasting key performance indicators, and how it can be used as a benchmark for a company.
- To provide an overview of time-series forecasting, including its definition, key concepts, and commonly used techniques.
- To demonstrate the application of time-series forecasting in a real-world context, using a real use case to illustrate its value as a tool for budgeting, decision-making and performance control.

2. THEORETICAL FRAMEWORK

A time series is a sequence of measurements of the same variable collected over time. Often, the observed values are equally distant in the time axis. Even though the target variable (y) might be continuous, the observed values, when projected on a discrete time axis (x), turn it into a sequence of discrete-time data (Weiss, 2018). The granularity of the time is not defined and so it may range from a higher granularity – milliseconds e.g. – to a lower granularity - years e.g. We can make use of time series in a very broad set of different fields such as healthcare, while monitoring patient vital signs over time like heart rate and blood pressure; on natural phenomena like temperature, ocean tides; and finance when tracking and analysing stock prices.

Before analysing, modelling or forecasting any given time series it´s of extremely importance of getting to know the data and its main characteristics. According to Chatfield, (2000) there are 4 main components to a time series:

a) Trend. It is loosely referred as “long-term change in the average level”. It is a long-term pattern of change in the data. It reflects the tendency for the target variable to take an upwards or downwards movement across time and can be easily identifiable depending on its strong or weak presence (Chatfield, 2000). Trends can be linear, meaning they have a steady and constant impact over time on the target variable; or they can be non-linear meaning its impact is more complex and non-constant over time. Logarithmic, exponential, and quadratic trends are examples of non-linear trends.
b) Seasonality. Seasonality in time series forecasting refers to a pattern of regular fluctuations that occur at fixed seasons within a year or other shorter period. These patterns may repeat annually, quarterly, monthly, weekly, or even daily (Chatfield, 2000). It can be caused by several factors such as weather, holidays, cultural events, or business cycles. Commonly, we can expect the target variable of a seasonal time-series to have higher or lower values on a specific season.
c) Cyclic variations. Unlike seasonal variations, these occur for a time span of at least one year (Chatfield, 2000). They are usually associated with macroeconomic factors and may be regular or non-periodic. One full iteration is called a cycle.
d) Noise. The noise component refers to the random fluctuations or irregularities that are present in the data and cannot be explained by the underlying patterns or trends (Chatfield, 2000). This component is sometimes referred to as the "error term" in statistical models and

it is an important consideration in time series analysis which can have significant implications for forecasting and prediction.

The following is a visualization of real time-series decomposition using BST Revenue data.

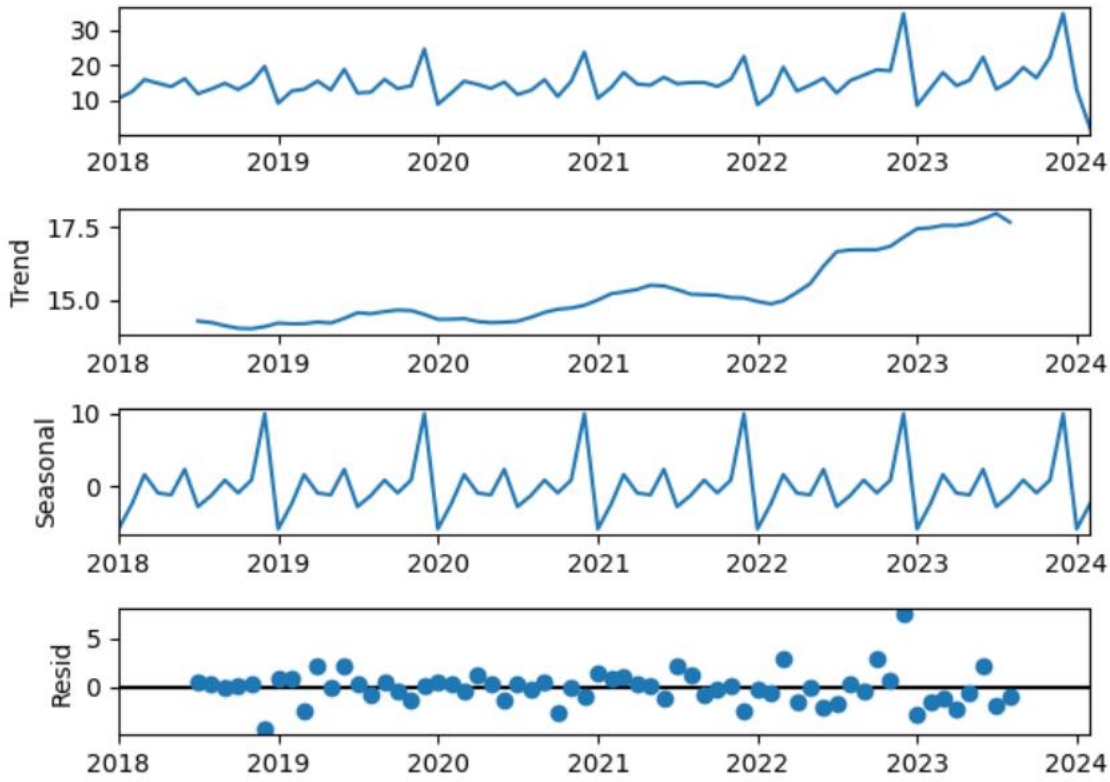

Figure 1 - SGB Revenue time-series decomposition

Therefore, it is possible to describe a time series through an Additive model:

$$X_t = T_t + S_t + C_t + e_t \quad (1)$$

or through a Multiplicative model:

$$X_t = T_t \times S_t \times C_t \times e_t \quad (2)$$

where X_t (t = 1,2… n) is the observed time-series, T_t is the Trend, S_t is the Seasonality, C_t is Cyclic Variations and e_t is the Noise (Nwogu, Iwueze, Dozie, & Machu, 2019).

The oldest time series models were developed in the beginning of the twentieth century and paved the way for more sophisticated techniques, such as autoregressive integrated moving average (ARIMA) and data smoothing, which are widely used today in fields such as finance, economics, and engineering (Gauta & Singh, 2020). Time series forecasting is an art firmly grounded in mathematical and statistical theory, but it is often based on empirical studies using benchmark methods or models as baseline (Gooijer & Hyndman, 2006). In practice, forecasters must navigate the complex aggregation of model assumptions and data limitations, carefully adapting their techniques to account for volatility and uncertainty in the data. Stationarity is a fundamental concept in time series analysis. It is a property of a time series if some of the most important statistical properties such as the mean, variance, and covariance remain constant over time, irrespective of the specific time interval being considered (Jalil & Rao, 2019). In other words, a stationary time series is one who's the mentioned above statistical properties do not depend on the specific point in time being analysed (Jalil & Rao, 2019). Forecasting models that rely on the assumption of stationarity can produce inaccurate predictions when applied to non-stationary time series data, leading to inefficient and biased forecasts (Saadi & Rahman, 2006). Statistically, a time series is stationary if the following characteristics apply: Constant mean over time: $E(X_t) = \mu$; Constant Variance over time: $Var(X_t) = \sigma^2$ and Constant Covariance over time: $Cov(X_t, X_{t-h}) = \alpha\,\sigma^2$, where μ, σ, α and σ are constants independents of t. To assess data stationarity, Dickey and Fuller proposed the Dickey-Fuller test for stationarity. There are different versions of the test and even a more complex test deriving from it called the Augmented Dickey-Fuller test, but they are based on the idea that testing for non-stationarity is the same as testing for unit root. It is based on equation (1) and it states that the under the null hypothesis of non-stationarity a unit root exists, $\delta = 1$ and, alternatively we have $\delta < 1$ and a stationary process (Tinungki, 2019). However, the test is not very powerful and does rely on assumptions like the normality of the residuals. This assumption is a common requirement in many time series forecasting models. If it is verified, then it allows statistical inference - statistical powerful methods such as hypothesis testing and confidence interval estimation. Contrarily, if it does not verify, then it is a sign of an incomplete model which is not being modelled with all the required information and is under performing, directly causing biased coefficients, and inflated standard errors (Maas & Hox, 2003). The normality of the error term assumption implies that it follows a normal distribution, with mean=0 and constant variance, $u \sim N(0, \sigma^2)$. It is an empirical question and, in some cases, it is very hard to proof undoubtedly and instead, one should treat it as an approximation to the theoretical assumption.

Quantile-quantile plots (Q-Q plots) can be used to visualize the residuals distribution and to assess error normality. They are plots of sample statistics against some expected quantiles from a standard normal distribution (Thode, 2002). The underlying theory behind the plot is that the residual normality test can be performed visualizing distribution of data. This means that if the points are close to the diagonal line, and no data is located far from the theoretical data distribution then we have a suggestion that the residuals are normally distributed (Alita, Putra, & Darwis, 2021). The general point of view is that we want to use statistical methods that are suggestive and constructive rather than formal procedures which are to be applied in the light of a tightly specified mathematical model which can be impossible to verify (Wilk & Gnanadesikan, 1968). So, even though the plot is not a formal test it can be used as an non-formal auxiliar text to assess model formulation.

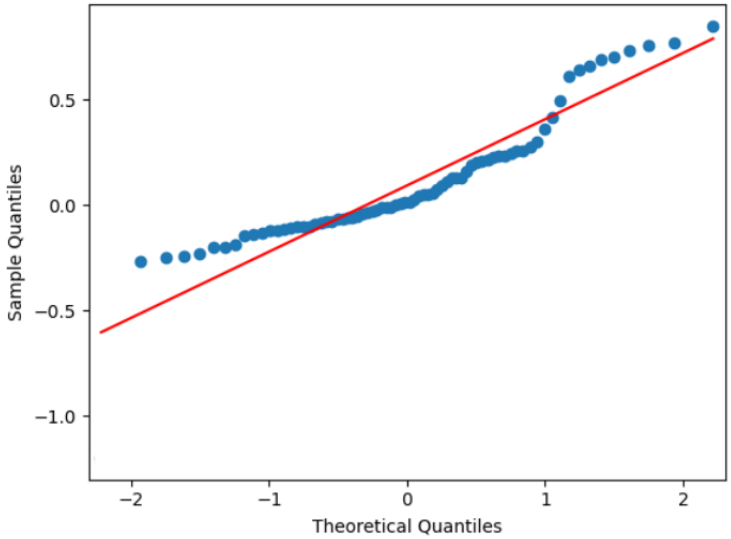

Figure 2 - Q-Q Plot of residuals

2.4.1. SARIMA

The Autoregressive Integrated Moving Average (ARIMA) models, first proposed by Box and Jenkins in 1976 (apud. Jain, Sharma, & Jain, 1985), have become a popular time series analysis technique for modelling and forecasting a wide range of phenomena. ARIMA models are a category of linear models that incorporate lagged values of the dependent variable to capture and model temporal dependencies, trends, and seasonality in time series data. The model can be broken down into pieces. The Auto-regressive component of an ARIMA model is the one which will be capturing and modelling the linear relationship between the target variable and the past values of itself (Reinsel, 2003). The number of past values to be included

will determine the Auto-regressive order of the equation. The simplest AR model is the AR(1) where only the last value of the target variable will be included. It can be written as:

$$X_t = \delta + \emptyset_1 X_{t-1} + w_t \quad (3)$$

where X_t (t = 1,2… n) is the observed time-series, δ is a constant, w_t is the residual and $\emptyset_n$(n = 1,2… *p)* is the auto-regressive coefficient of X_{t-n} .

Similar to the auto-regressive component previously described, the moving-average component of an ARIMA model will be capturing and modelling the linear relationship between the target variable and past errors. A moving average term is one which includes a past error and once again, the number of past error terms to be included will determine the Moving-average order of the equation. Moving average methods may appear in various forms, but the purpose remains the same, that is to track the trend determination of the given time series data (Hansun, 2013). The simplest MA model is the MA(1) where only the last error term will be included. It can be written as:

$$X_t = \mu + \theta_1 w_{t-1} + w_t \quad (4)$$

where X_t (t = 1,2… n) is the observed time-series, μ is a constant, w_t is the residual and θ_n(n = 1,2… *q)* is the moving average coefficient of the w_{t-n} .The SARIMA model is an extension to the classical ARIMA model which allows to model both trend and seasonality simultaneously (Bravo & Coelho, 2020). A Sarima model can be described as: *SARIMA (p,d,q)(P,D,Q)s*, where *p* is the Auto-regressive order, *i* is the Differencing order, *q* is the Moving Average order, *P* is the Seasonal Auto-regressive order, *D* is the Seasonal Differencing order, *Q* is the Seasonal Moving average order and *S* is the time span of repeating seasonal patterns. The seasonal auto-regressive component of a SARIMAX model is equivalent to the auto-regressive component except a seasonal interval is also applied. This means the lag value of the target variable to be included in the equation is determined by the seasonal interval. A seasonal ARIMA$(1,0,0)(1,0,0)_{12}$ can be written as:

$$X_t = \delta + \emptyset_1 X_{t-12} + \emptyset_1 X_{t-1} + w_t \quad (5)$$

Again, similarly, the seasonal moving-average component of a SARIMAX model is equivalent to the moving-average component except a seasonal interval is also applied. This means the lag value of the error term to be included in the equation is determined by the seasonal interval. A seasonal ARIMA$(0,0,1)(0,0,1)_{12}$ can be written as:

$$X_t = \mu + \theta w_{t-12} + \theta w_{t-1} + w_t \quad (6)$$

One of the most important assumptions of a SARIMA model is the assumption of Stationarity. That is the reason the *I* exist in SARIMA: it means integration but is commonly referred as differentiation, and, according to Box and Jenkins, the value of both *i* and *I* should not be greater than 2 (Park, Hong, & Jeon, 1984). The ACF and PACF plots are very useful time-series analysis resources that shows how linearly related two pairs of observations are at different time lags. They can be used to define the optimal values of a ARIMA model correlogram, but also to evaluate data stationarity (Alharbi & Csala, 2022): If the ACF plot slowly decay to zero and the PACF is significantly different in lagone, it means that the time-series is showing signs of non-stationarity (Tinungki, 2019).

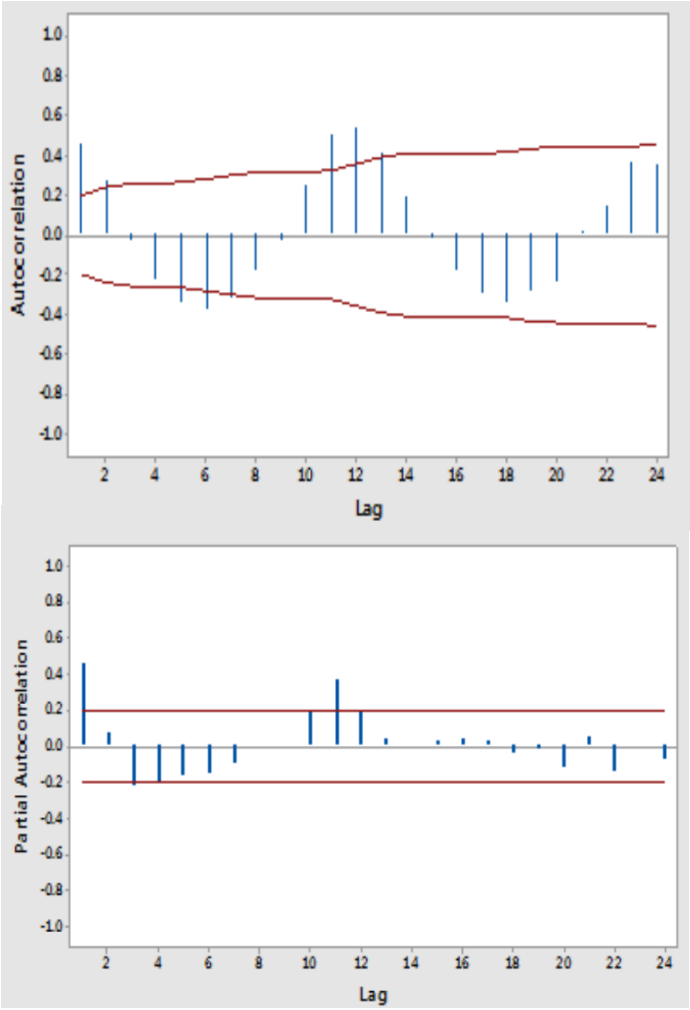

Figure 3- ACF and PACF of time-series X (Tinungki, 2019)

To overcome that, Box and Jenkins (apud. Barthélemy & Lubrano, 1995) propose differencing the time-series as a method do make it stationary. SARIMA models frequently utilize both the common differentiation as mentioned above as well as the seasonal differentiation. The 1st differenced process Y , is obtained from X as:

$$Y_t = X_t - X_{t-1} \tag{7}$$

And the 1st seasonal differenced process Y , is obtained from X as:

$$Y_t = X_t - X_{t-1-s} \tag{8}$$

Now, the ACF and PACF plots of the differenced time-series, Y_t exhibit a different behaviour, not retaining the previous mentioned characteristics which is indicative of a stationary time-series, which is, again, trivial to achieve a well specified model.

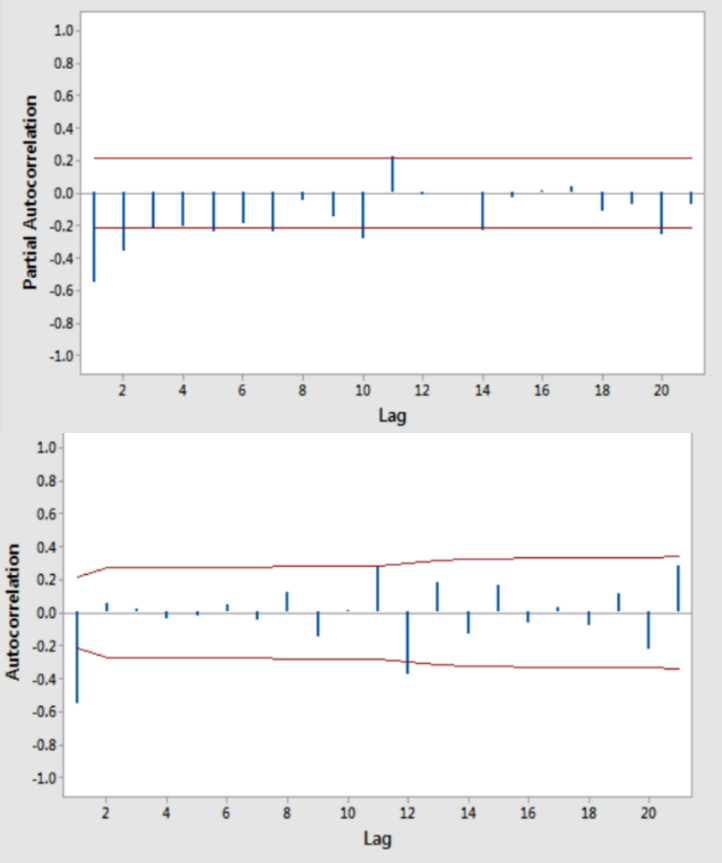

Figure 4 - ACF and PACF of differenced time-series X (Tinungki, 2019)

2.4.2. PROPHET

Prophet is an open-source time-series forecasting algorithm designed by Facebook. It was initially designed to handle common features of business time-series with features that can be intuitively tuned without the need to understand the underlying details of the model (Taylor & Benjamin, 2017).

It is based on an additive decomposable time-series model:

$$y(t) = g(t) + s(t) + h(t) + e_t$$

Where:

- g(t) is the trend function which models non-periodic changes in the value of the time-series.

- s(t) is the seasonality function which models periodic changes in the time-series.

- h(t) is the holidays function which represents the effect of known holidays, which might cause potential irregular changes.

- e_t is the error term.

Facebook´s Prophet may make use of two different categories of trend models: it can model a non-linear trend, which makes use of the logistic growth model; and a linear trend which includes a piece-wise constant rate of growth. The changepoints are key points where the growth rate is allowed to change (Taylor & Benjamin, 2017), which means that the trend function will be different on different points in time allowing prophet to model different trends accordingly with the changing characteristics of the data. In the following example image the change points are identified by the vertical red lines which generate different growth rate changes between them.

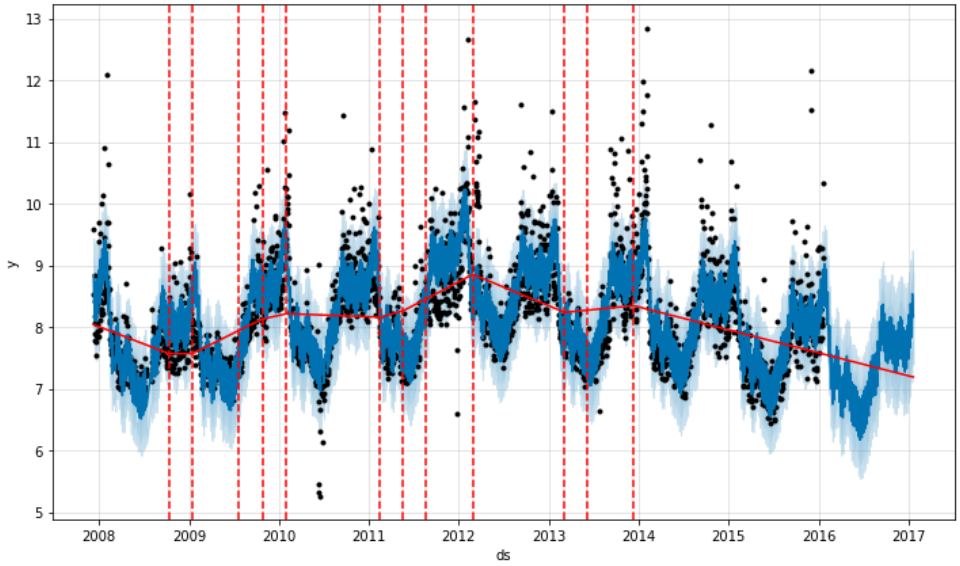

Figure 5- Trend changepoints detection (Taylor & Benjamin, 2017)

The changepoints can be automatically detected using a initial set of plausible candidates which will shrink in size until the optimal set is completely defined or, alternatively, they can be set manually by the analyst making use of his business knowledge to specify rate changing events, enhancing the model (Taylor & Benjamin, 2017). There are some prophet features which manage the trend and changepoints of the model such as *change_point_prior_scale*, *changepoints* and *n_changepoints*. They allow to adjust the flexibility of the trend function, which in combination with the number and location of changepoints will capture the trend and the real changes in the trend function while still preventing over-fitting (Taylor & Benjamin, 2017)..

Prophet makes use of the Fourier Series to model seasonality. The Fourier series describes the fluctuation of time series in terms of sinusoidal behaviour at various frequencies (Iwok, 2016). Moreover, business time-series often have multiple seasonalities because of the human behaviour they represent (Taylor & Benjamin, 2017), and that is why Prophet allows the inclusion of different types of seasonality simultaneously ranging from daily seasonality to yearly seasonality. The seasonality function can be controlled using prophet parameters such as *seasonality_prior_scale* and *seasonality_fourier_order* which will adjust both the complexity of the captured seasonality, i.e. the order of the Fourier series; as well as its weight on the final forecasts' calculation.

The holidays' function is meant to model specific events which are not specifically scheduled so can´t be captured with a usual seasonality function. Incorporating a list of holidays

into the model is made straightforward by assuming that the effects of holidays are independent (Taylor & Benjamin, 2017) and assigning to each holiday a parameter which reflects the corresponding change in the forecasts.

2.4.3. MODEL EVALUATION

Different criteria such as forecast error measurements, execution speed, interpretability and others have been used to assess the quality of a forecast (Shcherbakov, et al., 2013). When dealing with multiple objects, forecast error measurements are typically used for estimating the quality of forecasting methods and for choosing the best forecasting mechanism. A broad range of error measurements exist when dealing with any machine learning problem and time-series forecasting problems are not an exception. They can be divided into groups which include measures based on percentage errors, scaled errors or absolute forecasting error measures. (Shcherbakov, et al., 2013). Each measure has its drawbacks and the most suitable depends on the context of the problem we are trying to solve. Mean Absolute Error is a popular error measure and is calculated as follows:

$$\mathrm{MAE} = \frac{1}{m}\sum_{i=1}^{m} u_i \tag{9}$$

Where u_i (i = 1,2… n) is the residual at period *i*. It is not suitable if measuring objects with different scales or magnitudes because it is not able to relativize the error as well as it is not robust to outliers as extreme values will have a big impact on the final value (Shcherbakov, et al., 2013). However, it is very simple and easy to interpretate and a competitive option when measuring accuracy performance which makes it a reliable measure (Shortridge, Guikema, & Zaitchik, 2016).

3. METHODOLOGY

3.1. DATA

The BST's forecasting part is solely focused on SGB´s Revenue. SGB is a very big Siemens department which is already established in the market. Its cash flow is stable with a very accentuated seasonality. Revenue may come in any form from services, products, contracts, deals, projects, etc. For the time being we have access to historical data since 2018.

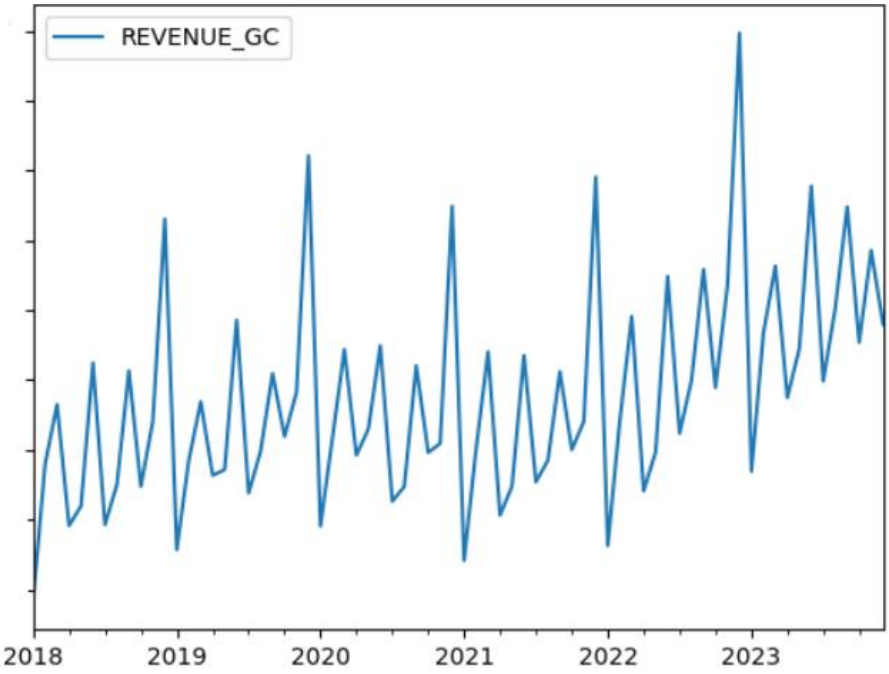

Figure 6- BST Revenue

The Order backlog is, similarly to the Revenue, a very important KPI. It reflects the value of incoming or scheduled deals. It is a very good indicator of how the business will develop on the following months or even years. For that reason, and even though it does not make it into the forecasting scope of BST, it can still be used as an explanatory variable and be included in the time-series forecasting models.

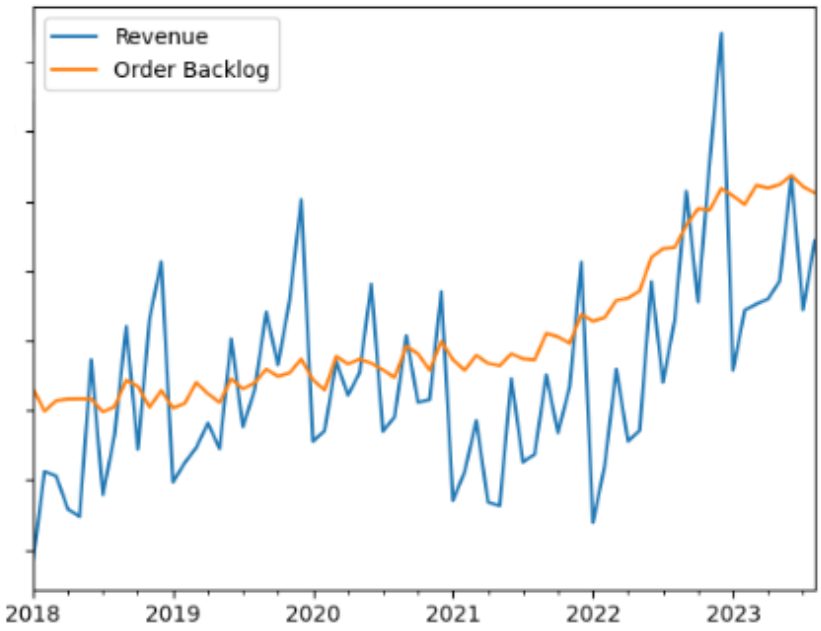

Figure 7- Revenue and Order Backlog

The data can be mapped to different segments, units or departments of the SGB organization. These organisational branches are distinguished by variables called IDENTIFIERS which include Country, District, Category, Range, Type, Store, and others which generate different levels of forecast granularity. They will have an important role for the forecast accuracy not only, but also, because some of those are more important than the others, as will be explained later in this document.

Figure 8- Data Granularity

The input data available for the BST lies in the HANA database, which is continuously updated with historical already distinguished by the above-mentioned IDENTIFIERS. This data is then ingested to a python environment in the AWS – Amazon Web Services, more specifically *AWS Sagemaker* where a *Jupyter* notebook instance exists solely for this purpose. This is where the BST algorithm takes place, and where all the typical steps from a Machine Learning pipeline are executed, resulting in the final forecasts. Once the python work is done and the predictions are calculated, the data is again sent to HANA, where it will be picked up by the front-end software Qlik Sense and is ready for consumption by the BST users.

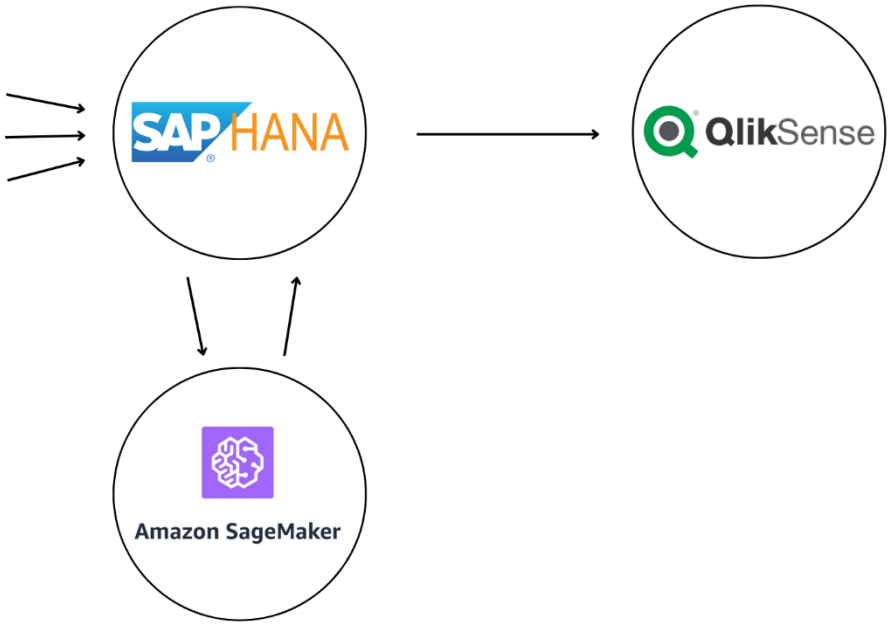

Figure 9 - Data Lifecycle

The BST runs every month using its newly ingested data from the HANA back end. Therefore, the data is transformed into monthly data points and the x-axis or time axis of all the existing time-series is composed of months. However, the forecasts will be available in the Qlik Sense front end both in monthly granularity as well as quarterly granularity.

As mentioned before, the existing time series will cover different parts of the SGB business. Then, the data volume does vary between bigger and smaller businesses which will cause relatively big forecasting capacity discrepancies between them. The models may also make use of an external variable, a regressor. Including an exogenous variable in time-series forecasting can be hard if we do not have the future values of the exogenous variable. The variable in question is, as mentioned before, the SGB Order Backlog. It is also a very important KPI in Siemens for which the future values are unknown. It means it is not straightforward to input it

in the models because even if the data is available for the training, we would not have it for the future prediction, making the forecast unviable. Moreover, the Order Backlog raw data reveals to be very unstable and therefore very useless as an explicative variable: we know it sets the pace for the incoming business years, but we cannot say for sure how and when and this information cannot be extrapolated from the data, even more if the data is very volatile and noisy. Despite the above-mentioned difficulties, we do know that relationship between Order Backlog and Revenue is a lagged relationship, which means we might not need the future values of Order Backlog. Instead, can make use of the actuals we already know to train the models and produce forecasts. However, this means that feature engineering of some kind would be needed. "Informative features are the basis for data analytics. They are useful for describing the underlying objects" (Dong & Liu, 2018) p.2 and then we are not interested in modelling the volatile behaviour of Order Backlog and input noise into our models. Instead, we are interested in inputting what we know it is useful data and information. As mentioned, before we do not have the knowledge to specify how and when Order Backlog will impact Revenue, but we know that it will, and we need to engineer it in a way it can be taken be interpreted by what it means, removing the noise. Furthermore, it will have different impacts on the different time-series depending on its organizational origins. For all those reasons, we need to smooth out the behaviour of the regressor (Raudis & Pabarskaite, 2016).

Moving average smoothing is a technique do used to decompose a time-series into two components: one very volatile and other, less volatile. The less volatile component is the one we are interested in and is obtained as follows:

$$Y_t = \frac{1}{m} \sum_{i=t-m-s}^{t-s} Y_i \qquad (10)$$

where Y_i (i = 1,2… n) is the observed time-series at period i, m is the moving average window size and s is a lag shift. In this case, the window represents the overall mean of backlog entries, and the lag shift represents the time that new order backlog entries will take until they are transformed in revenue. And, more importantly, it also ensures that we do not need to create artificial future regressor values because we are able to use actuals from the past. Several moving average smoothing can be applied to order backlog making use of different specifications window and lag shift specifications. The following image shows a scaled image

of SGB Revenue and Order Backlog for a specific time-series, using different options of the regressor feature engineering.

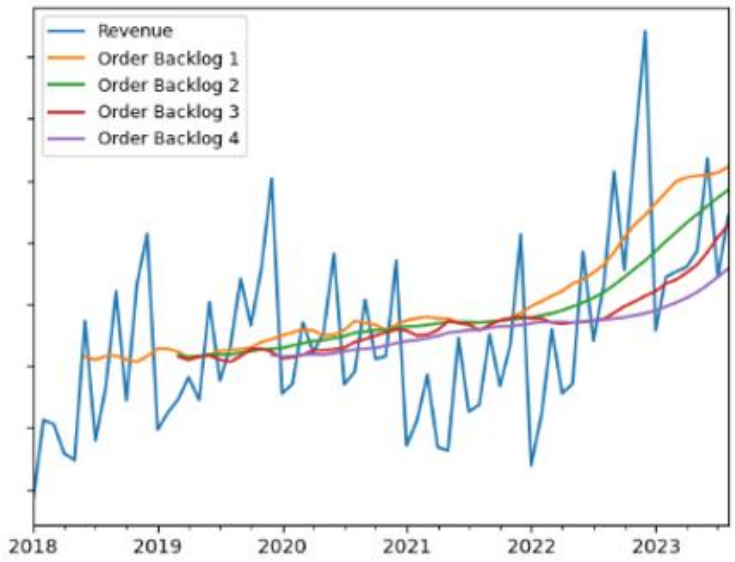

Figure 10- Revenue and Order Backlog feature engineering

It is clear in the image the different regressors are following a pattern along the time-series, setting the baseline for what Revenue can be expected on the following months, excluding. Shifting the line into the future, it can and will be used to allow an exogenous impact on the modelling.

3.2. MODELS

The BST model selection step is, unlike a usual time-series forecasting problem, a result orientated step. We are interested in continuously upgrading out forecasts by continuously searching for the best suitable model for each time-series every month as BST is refreshed and the new month´s version is created. Since the scope is so broad, and such a vast number of time-series forecasting models will be applied, validated, tuned and ranked, the BST forecasting approach needs to be generalized in such a way that its machine learning pipeline must suit every and each time-series, despite of their conceptual differences. Evaluating the performance of a predictive model is a trivial stage in any machine learning project. It is used to select the most appropriate model and its parameters, and it is one of the most reliable approaches to analyse the generalisation ability of predictive models (Cerqueira, Torgo, & Mozetic, 2020). Each one of the time-series is split into train and validation datasets as follows: the last year of

data is used for validation, and all the previous data points are used for training, which will usually result in nearly 60 data points for training, and 12 data points for validation. The data split can have relevant impact on forecasting performance and according to Medar, Rajpurohit, & Rashmi (2017) there might be a point where forecasting accuracy is lower due to not enough validation data even though training sample is large enough. Often, researchers might be unaware of the risks of applying more complex methods for model evaluation: cross-validation, when applied to time-series forecasting problems, could see the fundamental assumption of identically and independently distributed data violated even if the data was to be split randomly, due to the nature of the data which can be changing over time (Bergmei & Benítez, 2012). In this case, the validation dataset is aiming at an optimal mix between data representativeness on the validation set, as well as modelling power, maximizing the sample data to train the models. We want to have a short-term forecast (3 months-ahead forecast) and a long-term forecast (18 months-ahead forecast), and it is having that in mind that the models have distinct ways of calculating accuracy, for each different time-span horizon. The short-term model's validation score is calculated on the first 3 data points of the validation data, which correspond to the same three months that the short-term model will be forecasting. The long-term model is validated on the whole validation data set and will be predicting the remaining 15 months.

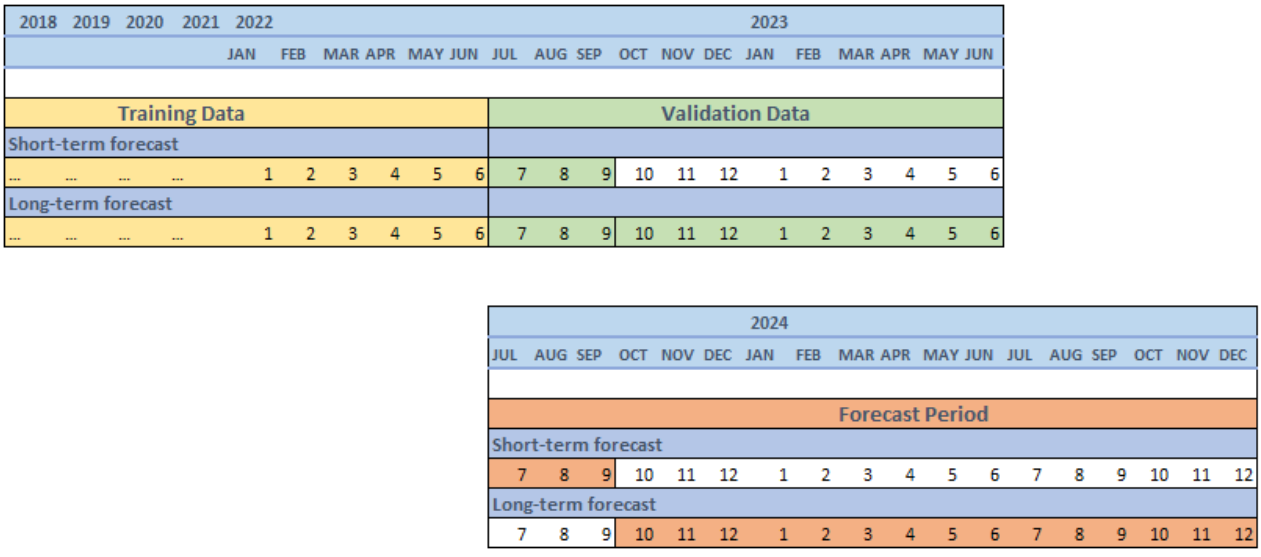

Figure 11- Train-Test Split

The training data is fed into the two machine-learning models available, in this case, the python libraries are: *statsmodels.SARIMAX* and *prophet.prophet*. They will make use of its corresponding parameter grid to search for its optimal specifications. This means, both the ARIMA model and the Prophet model go through an exhaustive parameter grid search to find

the optimal set of parameters. With the different parametrizations and accounting for both models, more than 1000 models are trained, and we will discuss those later. Each model´s performance is evaluated on the validation data, and they are ranked accordingly for each time-series, using MAE. Once the best models have been found, the best ranked model for each time-series will be again trained on all the historical data available and applied to produce the final forecasts.

The BST works on different levels of SGB´s organizational structure producing forecasts for the different levels which are distinguished by the 6 IDENTIFIERS previously described. It is then mandatory to be able to precisely map the higher-level revenue to the lower levels, whenever it is possible. If ignored, this means two or more different forecasts will be available for the same dimension and that is a problem because one could take the sum of all the lower-level forecasts and have a different forecast than the higher level forecast itself, which is not the purpose of BST as a SGB performance and controlling tool. For those reasons a bottom-up forecasting approach is utilized to ensure coherence along the different forecasting levels. Furthermore, this aggregation approach can be very useful for situations where the individual components of the forecast have specific behaviours or different patterns which is definitely the case for some of the levels we are dealing with (Lapide, 2006). In this case, it works in a way that if we have a forecast for the level X and 3 forecasts for the level X+1, the final forecast for the level X becomes the sum of the lower levels' forecasts and the higher-level forecast is ignored. However, this only applies if the forecasting level is complete. If the level is incomplete, meaning there is a time-series which does not have a forecast on the lower level, then the bottom-up forecasting does not apply to that level.

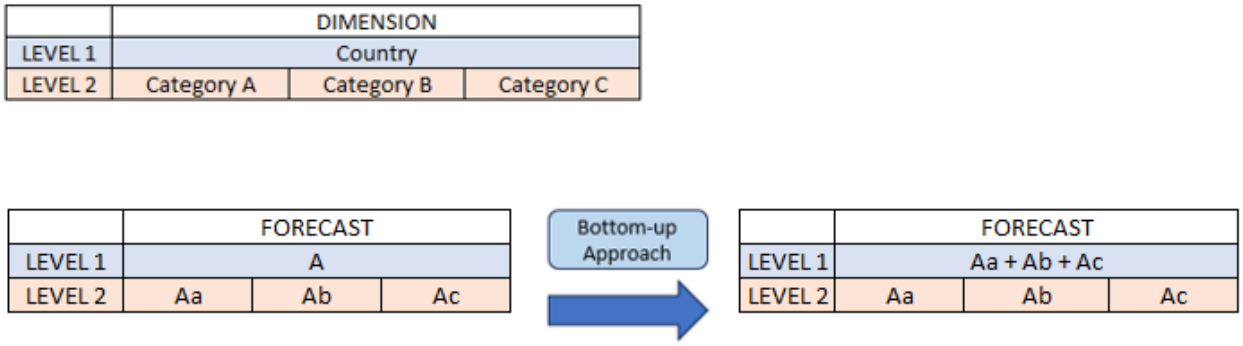

Figure 12- Bottom-up Forecast

4. RESULTS AND DISCUSSION

The BST is already an existing application in Siemens even before our paths cross. The BST exists only in Development environment: it is not a reliable and valuable tool for SGB because it still has severe flaws regarding forecast accuracy and integrity. It produces a very high quantity of forecasts simultaneously which makes its performance hard to control and monitor. One can describe it as very volatile: it is still very sensitive do data changes and data inconsistencies, either expected or unexpected.

SGB operates in several countries, and it is organized on very different levels distinguished by the mentioned IDENTIFIERS and other business branches. These have their own forecasting procedures, which are manually done by each entity and are available for use. Finally, BST is intended to produce competitive forecasts accuracy wise; to provide more detailed information diving deeper into the organization´s branches which do not have a manual forecasting procedure established; and to be a low-cost, low-effort, reliable and automatic tool that can be used for benchmarking by the SGB´s finance controlling sector.

In SGB environment, some kind of businesses are older, bigger, and more stable than others which make them also more predictable. That is easily verified at a naked eye even just observing the different time-series. On the following image we can see a relatively stable time-series.

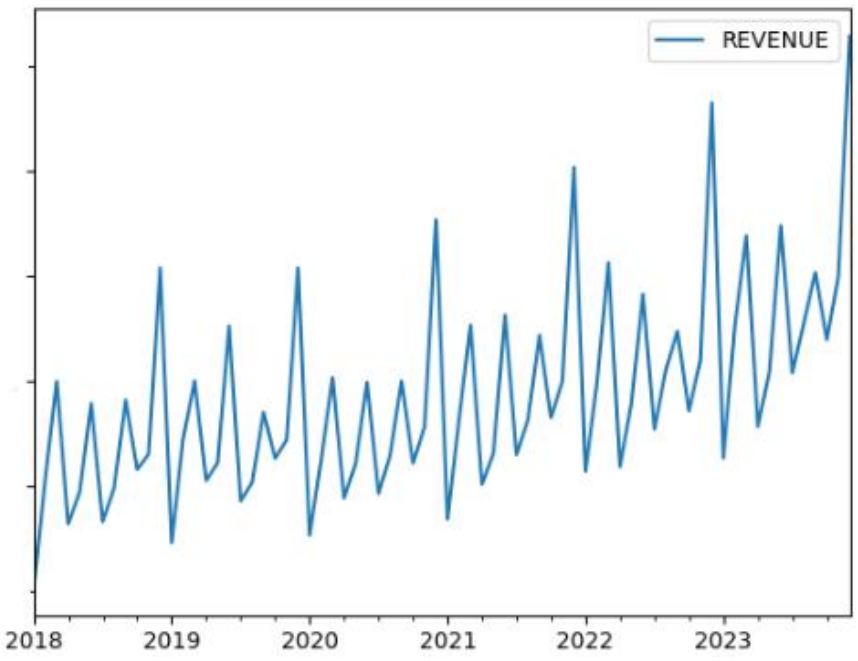

Figure 13- Stable business time-series

In contrast, a smaller time-series will usually display more volatile patterns which makes it interpretability and predictability harder. These will have impact on forecasting capabilities.

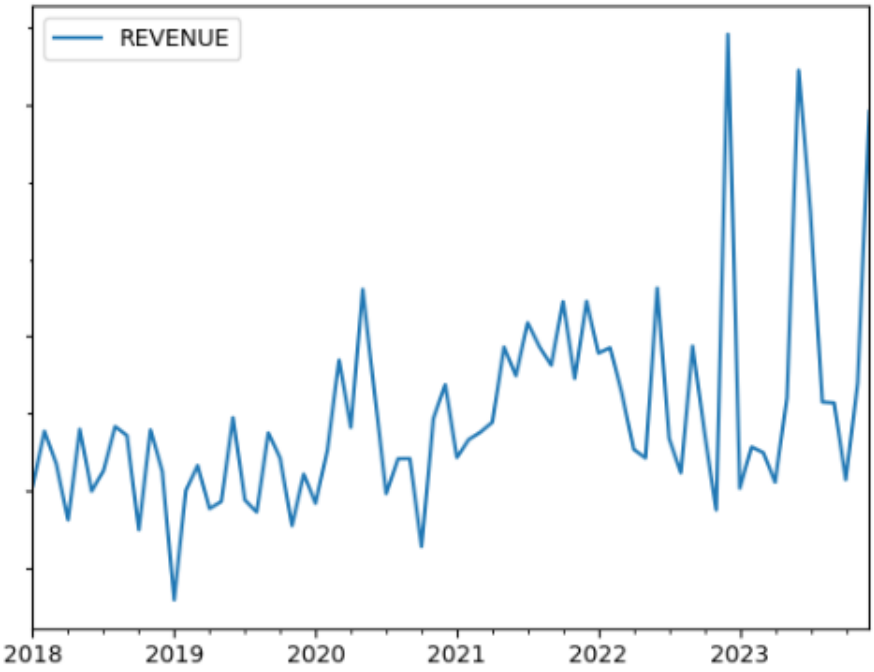

Figure 14- Unstable business time-series

To assess BST accuracy, we test its performance on all the twelve 2023 BST monthly versions: versions 202301- 202312 and assess its accuracy throughout the year focusing only on quarterly values – meaning forecast values for Q1, Q2, Q3 and Q4 are calculated and compared against the corresponding actuals. The following table shows BST performance on the first stages of its development: it includes 6 different levels of granularity, the 280 different possible models mentioned above in the document but no regressor is available for now. The figures are calculated using the average BST accuracy of each version on its remainder quarters and taking the overall average.

LEVEL	count	average	median
L1	33	12,96%	10,70%
L2	55	20,38%	12,70%
L3	127	41,19%	15,20%
L4	251	30,25%	18,30%
L5	643	63,40%	22,90%
L6	2184	49,70%	25,90%
TOTAL	3293	49,70%	23,90%

Table 1 - BST Accuracy a)

It is clear the more granular the forecasting, the less accurate it becomes which goes in line with the forecasting difficulties mentioned for smaller businesses. At this point the bigger forecasts show acceptable accuracy, but they are not the main character, the smaller forecasts´ performance is deal breaking for BST – it is not reliable enough. BST produces around 3300 forecasts, but it must not produce forecasts which show no integrity, misleading the users and hurting BST ´s reputation. At the highest granularity some forecasts show unacceptable levels of accuracy with MAE scores surpassing the 500% and resulting in an average of 49,7% MAE. Furthermore, due to the bottom-up approach we need to apply, those might even hurt the low granularity forecasts.

It is very hard to track performance on such a big number of forecasts – we need to identify strengths and weaknesses, define priorities, and tackle key improvement points. The highest granularity forecasting is not the priority, but they are holding us back from tackling the most relevant topics. For those reasons, this granularity of forecasting cannot be achieved for now and we must make changes. Forecasting identifiers 2 and 6 is not valuable enough to compensate the above issues and be kept in the BST scope. After its removal as well as some structural changes to the identifiers aligned with vision of management, we can re-assess BST performance. The accuracy changes are displayed next.

LEVEL	count	average	median
L1	33	11,20%	9,60%
L3	88	31,50%	14,90%
L4	180	54,20%	16,20%
L5	382	48,50%	20,60%
TOTAL	716	43,90%	17,30%

Table 2- BST Accuracy b)

Removing those levels of forecasting reduces substantially the total number of time-series from 3293 to 716 which is around 22% of the previous number. We already verify a performance improvement, and the most important part is the big shift in the total median accuracy: reducing from 23,9% to 17,3% is a 6,6 percentual points shift and represents a 27,6% improvement, which means that, even though the average accuracy is still fairly similar, the accuracy distribution is far less skewed to the left. This means that a vast number of extreme forecasts do not exist anymore and we should now focus on identifying and refining the key improvement points.

There are several aspects of a machine learning pipeline that can be optimized eyeing the best performance possible: hyperparameter tuning is one that stands out from the rest because finding the best parameters or specifications can result in a huge positive shift in performance. The ARIMA and Prophet models have several parameters that can be fine-tuned and that is the reason so many models are evaluated, so that we can test several different combinations. However, just like He, Wang & Jiang (2008) state, "parameters tuning is likely to become difficult for the reason that the parameters space sometimes is very complex.", and that is why there is a certain point where the parameter search becomes saturated, meaning the expansion of the parameter search does not result in better forecast performance and, in contrary it results only in extra computational effort and extra running time. Even if we would be able to find a better performing model on the validation set, the thin margins between the best models would mean that we cannot conclude on which one is the most appropriate without taking the risk of overfitting. But we are interested in mitigating the overfitting risk anyways, which can be done by manipulating the train and validation data sets - increasing or decreasing the validation set looking for the optimal split, both for the short-term performance and for the long-term performance. However, again, it is hard to evaluate performance on such a big number of time-series which makes it hard to compare approaches that have thin performance differences which means we might need to decide beforehand what is a good enough performance. Other key aspect is the inclusion of exogenous variables: the Order Backlog really should be a great indicator to estimate upcoming revenue. It has always been a subject of great interest: but how can it be used by BST to finally take its forecasting accuracy to the next level? Several ways of including a regressor in the models have been tested mostly ranging between the options of producing a regressor forecast or the regressor feature engineering mentioned before. But forecasting the regressor is as complicated of a problem as forecasting revenue and it has other drawbacks – it introduces extra uncertainty and volatility to the forecasts making some forecasts´ integrity questionable which is an issue we had and overcame before. Moreover, accuracy wise, along with the difficulties associated with forecasting a regressor it might still not even be desired considering the lagged relationship between the two variables that we have covered before.

In the end, we decide to use both the data split and the regressor as an engineered regressor like explained in Section 3, which combined with the parameter search result in 1440 possible options.

ARIMA		Prophet		Regressor(lag shift, window size)	
p	[1,2]	changepoint_prior_scale	[0.05, 0.25, 0.5, 0.75	No Regressor	
i	[0,1]	seasonality_period	['yearly']	Order Backlog MA (3,3)	
q	[0,1,2]	seasonality_prior_scale	[3, 5, 7, 10]	Order Backlog MA (3, 18)	
P	[1,2]	fourier_order	[0.05, 0.75, 10]	Order Backlog MA (18, 3)	
I	[0,1]	regressor_prior_scale	[0.1, 1, 10]	Order Backlog MA (18, 18)	
Q	[0,1,2]				
S	[12]				
Total	144	144		x5	1440 models

Table 3 - BST available models

And, re-evaluating BST performance, these are the accuracy figures:

LEVEL	count	average	median
L1	33	10,70%	9,10%
L3	88	27,20%	15,10%
L4	180	29,60%	15,40%
L5	382	35,60%	19,50%
TOTAL	716	30,20%	16,50%

Table 4- BST Accuracy c)

Once again, we see a substantial improvement on performance, this time on the average accuracy. Reducing the total accuracy from 43,9% to 30,2% means 13,7 percentual points and a 31% improvement. This is a key achievement as the forecasts are getting more and more accurate and BST is taking steps in the right direction.

This shift in performance means the BST is now very competitive with the manual forecasting procedures. These procedures are not as granular as BST – they only go as deep as LEVEL 2 which means that is the deepest level we can compare, but it is also the most important levels from a finance perspective and, therefore, from a management perspective, since they will include the biggest time-series. Moreover, following the same reasoning, some countries are most important than others because they are the core of SGB business and performance on these levels are key thresholds to define BST ´s success. The following table shows the average BST performance when compared against the manual forecasting. Some minor time-series have been excluded from this analysis.

	TOP 5 countries		Other countries	
LEVEL	BST	MFC	BST	MFC
L1 - Country	2,50%	1,80%	8,70%	9,50%
L3 - Category A	4,70%	5,50%	15,10%	19,70%
L3 - Category B	3,30%	3,30%	7,50%	8,60%
L3 - Category C	6,80%	9,10%	36,20%	36,80%

Table 5- BST vs MFC

We see BST is capable of forecasting with a level of accuracy very similar to the manual forecast, even outperforming it when looking at overall performance across all countries. And being extremely competitive in the top 5 countries where forecast accuracy is already very high achieving values around the 2% MAE at country level and slightly higher values at Category Level. This is a huge achievement for BST because it means its performance is good enough from a management perspective and it has proven itself as a valuable budgeting and performance controlling tool and is now ready to be launched in Production environment.

5. CONCLUSION

Even though the project is successful in the end, several obstacles have aroused during the process: the historical data is only available since 2018 which means only around 60 data points exist, in the best-case scenario. In other cases, even less data points which sometimes can be troublesome for the forecasting models which lay on statistical properties that need a certain sample size. Moreover, this small sample size does not allow a lot of flexibility when engineering variables, testing different data sets. Furthermore, more complex models like deep learning forecasting models also require more data flexibility. The broad scope of BST joint with the automatism and low time-effort required also means we need to have a very result orientated algorithm and need to leave more qualitative analysis of the time-series apart which might cost both transparency and performance. BST is continuously updated with new data, so its forecasts are in constant improvement which means it also needs constant monitoring. More concretely, the machine learning pipeline must be sporadically re-evaluated in the search for the best approach as new ideas or requests may arrive from the customers or the management itself, but one should always have a strictly defined step-by-step guide of which development is to be done to achieve which goal or else he or she may lose track and end up with a very inefficient or ineffective work.

The project kicked off in November 2022, and that is where I started to get to know the product by its data sources which also introduce me to both the Siemens and the SGB environment and allowed me to learn more on their business practices and, more specifically, allowed me to acknowledge the benefits BST could bring as an automatic, low-effort and low-cost application. It is very interesting to being able to apply concepts learned during both my master thesis and my bachelor´s degree in NOVA IMS in a real word use case. These include data processing; data analysis; machine learning; and other field specific concepts like time-series analysis and ARIMA models. And also learn that more and more applications can be born from these concepts taking into account the big recent spike in interest and development of artificial intelligence. The journey was very long but it is very rewarding to see the work developed paying off as BST has become an important tool in the SGB environment with users from all around the globe.

6. BIBLIOGRAPHY

Alharbi, F. R., & Csala, D. (2022). A Seasonal Autoregressive Integrated Moving Average with Exogenous Factors (SARIMAX) Forecasting Model-Based Time Series Approach. *Inventions*, *7*(4). https://doi.org/10.3390/inventions7040094

Alita, D., Putra, A. D., & Darwis, D. (2021). Analysis of classic assumption test and multiple linear regression coefficient test for employee structural office recommendation. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, *15*(3), 295. https://doi.org/10.22146/ijccs.65586

Bergmeir, C., & Benítez, J. M. (2012). On the use of cross-validation for time series predictor evaluation. *Information Sciences*, 191, 192–213. https://doi.org/10.1016/j.ins.2011.12.028

Cerqueira, V., Torgo, L., & Mozetic, I. (2019). *Evaluating time series forecasting models: An empirical study on performance estimation methods*. https://doi.org/10.1007/s10994-020-05910-7

Chatfield, C. (2000). *Time-series forecasting*. CRC press.

Coelho, E., & Bravo, J. (2020). *Short_Term_Regional_Demographic_Forecasts*.

Coveney, M., & Cokins, G. (2014). *Budgeting, planning and forecasting in uncertain times*. https://egrove.olemiss.edu/aicpa_guides

De Gooijer, J. G., & Hyndman, R. J. (2006). 25 years of time series forecasting. *International Journal of Forecasting*, *22*(3), 443–473. https://doi.org/10.1016/j.ijforecast.2006.01.001

Dong, G., & Liu Huan. (2018). *FEATURE ENGINEERING FOR MACHINE LEARNING AND DATA ANALYTICS*. https://www.crcpress.com/Chapman--HallCRC-Data-Mining-and-Knowledge-Discovery-Series/book-series/CHDAMINODIS

Ensafi, Y., Amin, S. H., Zhang, G., & Shah, B. (2022). Time-series forecasting of seasonal items sales using machine learning – A comparative analysis. *International Journal of Information Management Data Insights*, *2*(1). https://doi.org/10.1016/j.jjimei.2022.100058

Estatística, I. N. (2020). Short term Regional Demographic Forecasts with Time Series Methods and Machine Learning Algorithms.

Gautam, A., & Singh, V. (2020). Parametric versus non-parametric time series forecasting methods: A review. In *Journal of Engineering Science and Technology Review* (Vol. 13, Issue 3, pp. 165–171). Eastern Macedonia and Thrace Institute of Technology. https://doi.org/10.25103/JESTR.133.18

De Gooijer, J. G., & Hyndman, R. J. (2006). 25 years of time series forecasting. *International journal of forecasting*, *22*(3), 443-473.

Hansun, S. (2013, November). A new approach of moving average method in time series analysis. In *2013 conference on new media studies (CoNMedia)* (pp. 1-4). IEEE.

He, W., Wang, Z., & Jiang, H. (2008). Model optimizing and feature selecting for support vector regression in time series forecasting. *Neurocomputing*, *72*(1–3), 600–611. https://doi.org/10.1016/j.neucom.2007.11.010

Iwok, I. A. (2016). Seasonal Modelling of Fourier Series with Linear Trend. *International Journal of Statistics and Probability*, *5*(6), 65. https://doi.org/10.5539/ijsp.v5n6p65

Jain, R. K., Sharma, R. D., & Jain, S. (1985). Application of ARIMA model in adjustment of seasonal and non-seasonal variations in births of Ontario. *Genus, 41(3-4), 127–133.*

Kahn, K.B. (1998). Revisiting top-down versus bottom-up forecasting. *The Journal of Business Forecasting*, 17, 14-19

Maas, C. J. M., & Hox, J. J. (2004). The influence of violations of assumptions on multilevel parameter estimates and their standard errors. *Computational Statistics and Data Analysis*, *46*(3), 427–440. https://doi.org/10.1016/j.csda.2003.08.006

Medar, R., Rajpurohit, V. S., & Rashmi, B. (2017, July 2). Impact of Training and Testing Data Splits on Accuracy of Time Series Forecasting in Machine Learning. *2017 International Conference on Computing, Communication, Control and Automation, ICCUBEA 2017*. https://doi.org/10.1109/ICCUBEA.2017.8463779

Nwogu, E. C., Celestine, K., Dozie, N., Ifeyinwa, M. H., Nwogu, E. C., Iwueze, I. S., Dozie, K. C. N., & Mbachu, H. I. (2019a). Choice between Mixed and Multiplicative Models in Time Series Decomposition. *International Journal of Statistics and Applications*, *2019*(5), 153–159. https://doi.org/10.5923/j.statistics.20190905.04

Libert, G., (1984), An automatic procedure for Box-Jenkins model building, *European Journal of Operational Research, 17, issue 1, p. 95-103.*

Passoja, P. (2015). *BUDGETING AND FORECASTING APPLICATION DEVELOPMENT AND EVALUATION*.

Raudys, A., & Pabarškaitė, Ž. (2018). Optimising the smoothness and accuracy of moving average for stock price data. *Technological and Economic Development of Economy*, *24*(3), 984–1003. https://doi.org/10.3846/20294913.2016.1216906

Reinsel, G. C. (2003). Elements of multivariate time series analysis. *Springer Science & Business Media.*

Saadi, S., & Rahman, A. (2008). Evidence of non-stationary bias in scaling by square root of time: Implications for Value-at-Risk. *Journal of International Financial Markets, Institutions and Money*, *18*(3), 272–289. https://doi.org/10.1016/j.intfin.2006.12.001

Shcherbakov, M. V., Brebels, A., Shcherbakova, N. L., Tyukov, A. P., Janovsky, T. A., & Kamaev, V. A. evich. (2013). A survey of forecast error measures. *World Applied Sciences Journal*, *24*(24), 171–176. https://doi.org/10.5829/idosi.wasj.2013.24.itmies.80032

Shortridge, J. E., Guikema, S. D., & Zaitchik, B. F. (2016). Machine learning methods for empirical streamflow simulation: A comparison of model accuracy, interpretability, and uncertainty in seasonal watersheds. *Hydrology and Earth System Sciences*, *20*(7), 2611–2628. https://doi.org/10.5194/hess-20-2611-2016

Taylor, S. J., & Letham, B. (2017). *Forecasting at Scale*. https://doi.org/10.7287/peerj.preprints.3190v2

Thode, H.C. (2002). Testing For Normality. *CRC Press*. https://doi.org/10.1201/9780203910894

Tinungki, G. M. (2019). The analysis of partial autocorrelation function in predicting maximum wind speed. *IOP Conference Series: Earth and Environmental Science*, *235*(1). https://doi.org/10.1088/1755-1315/235/1/012097

Weiß, C. H. (2018). *An introduction to discrete-valued time series*. John Wiley & Sons.

Wilk, M.B. and Gnanadesikan, R. (1968) Probability Plotting Methods for the Analysis of Data. *Biometrika,*55,1-17. https://doi.org/10.2307/2334448