University Institute of Lisbon

Department of Information Science and Technology

Time Series Forecast and Anomaly Detection at Scale applied to Business Metrics in an ERP Environment

André Gil Cardeira Martins

Dissertation submitted as partial fulfillment of the requirements for the degree of

Master (MSc) in Business Intelligence Systems

Supervisor:
José Dias Curto, Associate Professor, PhD ISCTE-IUL Business School

September 2019

Acknowledgments

I want to express my thanks to Professor José Dias Curto for encouraging through all the investigation.

A very special thanks to my wife Elisabete and my sons Miguel and Inês, who gave me all the love, time and energy that allowed me to accomplish this work. To my mother Augusta, father Ramiro, and brother Ricardo for all the support. To my mother-in-law Lurdes and father-in-law Francisco, for all the help with Miguel and Inês every time it was needed.

I also want to thank Eng. José Dionísio and Eng. Jorge Baptista, for allowing me to develop this thesis at Primavera BSS. And to Miguel Domingues and all my colleagues from Innovation and New Technology Team for all the positive energy when the results were far away from being good.

And at last for a close circle of friends, for making me believe that I could accomplish this work.

To all I have listed, my sincere "Thank you."

Resumo

No meio empresarial, "*dashboards*" são mecanismos analíticos amplamente utilizados que ajudam no processo de tomada de decisão ao exibirem insights, indicadores de desempenho (KPIs) e métricas de negócio. A informação disponibilizada por este tipo de mecanismo é fortemente agregada, de forma a obter-se um elevado nível de sumarização e consequentemente facilitar a sua consulta. No entanto, a necessária sumarização provoca o surgimento de "*blind spots*", ao ocultar informação importante como, por exemplo, uma quebra acentuada de receita de um cliente, ou de um vendedor, ou de um produto/serviço específico. Estes "*blind spots*" dificultam a deteção de eventuais problemas e oportunidades de negócio, que ficam dependentes de uma exploração adicional demorada e minuciosa. Adicionalmente, o processo de transformação digital tem como consequência um aumento substancial do número de métricas referentes a todos os sistemas que suportam o negócio, que importa acompanhar. Desta forma, será possível antecipar ações baseadas na previsão de um comportamento futuro, bem como detetar um eventual desvio isolado ou sucessivo face ao seu comportamento espectável.

Como objetivo desta dissertação pretendemos promover a obtenção de conhecimento a partir de dados de negócio, através da aplicação de técnicas de Aprendizagem Automática ("Machine Learning"). Tendo por base o processo de tomada de decisão a partir de dados ("*Data-Driven Decision-Making*") pretende-se propor a integração numa aplicação ERP de um mecanismo que permita prever o comportamento futuro de séries temporais que contêm dados de negócio, bem como detetar e medir possíveis anomalias de forma a poderem ser gerados alertas.

Para lidar com uma ampla diversidade de séries temporais, propomos um método de previsão de meta-aprendizagem que utiliza um classificador para identificar o melhor método de previsão para cada série temporal, e uma nova métrica inteligente que permite ordenar séries temporais pela anomalia acumulada.

O conhecimento gerado irá complementar a informação disponibilizada pelos mecanismos analíticos tipicamente existente numa aplicação ERP (incluindo "*dashboards*"). Desta forma pretendemos contribuir para uma maximização dos proveitos e redução da possibilidade de erro ou fraude, bem como do desperdício e consequentemente mitigar a incerteza e diminuir o risco operacional.

Pretende-se igualmente que a solução promova a utilização de Aprendizagem Automática em Pequenas e Médias Empresas, e consequentemente uma futura implementação de tomada de decisões a partir de Inteligência Artificial ("*AI-Driven Decision Making*"), onde uma reação assertiva e automatizada é despoletada, face a problemas ou oportunidades encontradas, mas cujo estudo fica fora do âmbito do presente trabalho.

Palavras-Chave: Séries Temporais, Previsão, Deteção de Anomalias, Previsão em Negócios, Modelação de Incerteza.

Abstract

In the business world, dashboards are a widely used analytical mechanism that helps in the decision-making process by displaying insights, key performance indicators, and business metrics. The information provided by this type of mechanism is strongly aggregated, to obtain a high level of summarization and consequently make reading easier. However, the necessary summarization causes "blind spots" to appear by hiding important information such as a sharp drop in revenue from a specific customer, seller, or product/ service. These "blind spots" make it difficult to detect potential business problems and opportunities, which depend on lengthy and thorough additional exploration. Also, the digital transformation process has resulted in a substantial increase in the number of metrics for all systems supporting the business that need to be tracked. Thus, it will be possible to anticipate actions based on the prediction of future behavior, as well as to detect any isolated or successive deviation from the expected behavior.

With this dissertation, we intend to promote the acquisition of knowledge from business data through the application of Machine Learning techniques. Based on the Data-Driven Decision-Making process, we intend to propose integration into an ERP application of a mechanism to predict time-series behavior, as well as detecting and measuring possible anomalies.

For dealing with a wide diversity of time series, we propose a meta-learning forecasting method that uses a classifier to identify the best forecasting method for each time series. We also propose a new intelligent metric that allows us to sort time series by the accumulated anomaly.

The knowledge generated will complement the information provided by the analytical mechanisms typically present in an ERP application (including dashboards). In this way, we intend to contribute to the maximization of profits and reduction of the possibility of error or fraud, as well as waste and consequently mitigate uncertainty and reduce operational risk.

Our solution should promote the need to use Machine Learning in Small and Medium Enterprises, and consequently, future implementation of AI-Driven Decision Making. AI-Driven Decision-Making purposes an assertive and automated reaction to problems or opportunities encountered, but whose study is outside the scope of this dissertation.

Keywords: Time Series, Forecasting, Anomaly Detection, Business Forecasting, Uncertain Modeling

Contents

List of charts

List of figures

Acronyms

AI – Artificial Intelligence

ARIMA - Autoregressive Integrated Moving Average

BI – Business Intelligence

CPU – Central Processing Unit

DM – Data Mining

DS – Data Science

DSR – Design Science Research

DSRM – Design Science Research Methodology

ERP - Enterprise Resource Planning

IoT – Internet of Things

IT – Information Technology

IS – Information Systems

KPIs - Key Performance Indicator

MAPE - Mean Absolute Percentage Error

ML – Machine Learning

RQ – Research Question

PA – Predictive Analytics

POS – Point of Sale

SaaS – Software as a Service

SMAPE - Symmetric Mean Absolute Percentage Error

SMB - Small and Medium Business

SVM – Support Vector Machine

1 – Introduction

This chapter of the dissertation will present the purpose of the research and the research context, the chosen research methodology, research hypotheses as well as a brief description of the work structure

This chapter is organized in the following Sections:

- Section 1.1 – presents the purpose of the research
- Section 1.2 – presents the motivation and research context
- Section 1.3 – presents the research methodology
- Section 1.4 – presents the research hypotheses
- Section 1.5 – presents the document structure

1.1. Purpose of the Research

The forecasting process, in a business context, plays an important role in the development of any organization, influencing human and financial resources.

In (Taylor, et al., 2017 p. 1) forecasting is referred as "a common data science task that helps organizations with capacity planning, goal setting, and anomaly detection.". In (Hyndman, et al., 2019) Section 4.2, it is clarified the difference between forecasting, Goals, and Planning. "Forecasting is about predicting the future as accurately as possible." Goals "are what you would like to have happened," and Planning involves "determining the appropriate actions that are required to make your forecasts match your goals."

Decisions such as the opening of a new delegation, the need to hire new employees, or the elementary definition of an individual or collective sales goals, impacts on the cash flow of any organization. On the other hand, poor forecasting methodology can lead to unnecessary investment and waste or stock breach that will have a preponderant influence in any organization.

The propose of this research is to promote the use of Data-Driven Decision-Making (DDD), by generating knowledge-based on Business Metrics. The data-driven decision-making process the practice of basing decision-making on data analysis, as opposed to the exclusive use of intuition. A direct relationship between data-based decision making and company performance is demonstrated in (Brynjolfsson, et al., 2011). "Our results suggest that DDD capabilities can be modeled as intangible assets which are valued by investors and which increase output and profitability."

Technology companies and large-scale businesses are collecting a huge amount of time series data and are following their behavior closely. Examples of time series data in technology companies are web-click logs, web search counts, number of users in a specific service, etc. Examples in a typical business are sales, costs, the demand for products, etc.

Organizations use analytic tools like reports and dashboards to monitor the business performance. Dashboards are a widely used analytical mechanism that helps in the decision-making process by displaying insights, key performance indicators, and business metrics

If an organization that has one thousand active products, continuously analyzing the plotting of one thousand sales would be repetitive and time-consuming work. The commonly used solution is to aggregate results to obtain a high level of summarization. Nevertheless, aggregating information to KPIs generates what is called "blind spots," by hiding important information such as a sharp drop in revenue from a specific customer, seller, or product/ service.

Typically, dashboards allow doing a "drill-down analysis," where we can view detailed information that is influencing the global result of the KPI. As this functionality, appears to reduce the "blind spot" problem introduced before, it is easy to create a situation where the opposite extreme situation in a normal behavior in the KPI. Additionally, the increasing number of metrics and KPIs and the difficulty in defining the "normal behavior" end up corroborate the need for using other solutions for business analytic propose.

An Anomaly Detection system would create a list of incidents, which could be sort by a raking score, that compares the importance of the incident face the other incidents. The list of incidents would be outputted to someone who would analyze if it was a problem and if there's something to be done to fix it.

This work intends to create an autonomous forecasting and anomaly detection methodology. We will try to prove that forecasting and anomaly detection systems in an ERP environment can create value.

The possibility of applying automated solutions based on forecasting and anomaly detection is out of the scope of this work, and it's called Artificial Intelligence-Driven Decisions Making (AI-DDM) (Colson, 2019). The main idea in this work is to develop a tool to empower humans in the decision-making process.

As mentioned in (LUCA, 2019 p. 38), what "Algorithms capable of making predictions (…) can do is extremely powerful: identifying patterns too subtle to be detected by human observation, and using those patterns to generate accurate insights and inform better decision making. The challenge for us is to understand their risks and limitations and, through effective management, unlock their remarkable potential."

For dealing with a wide diversity of time series, we propose a meta-learning forecasting method that uses a classifier to identify the best forecasting method for each time series. We also propose a "Pri Anomaly" metric that is an accumulator of the

measure of local anomalies that allow us to highlight major accumulated anomaly time series.

1.2. Motivation and Research Context

The importance of demand forecasting is mentioned in (Goodwin, 2018 p. 2) to be "crucial to the operations of most companies." Some examples are "Inventory planning, logistics planning, production scheduling, cash flow planning, decisions on staffing levels, and purchasing decisions can all depend on forecasts." Author refers to the benefit of the use of good forecasting solutions "to improve customer service levels and so foster customer goodwill and retention and lower costs.". On the other hand "there will be less need for expensive emergency production runs, and there should be a reduction in the waste associated with excessive stock levels and unsold products."

The authors refer two examples: "One forecasting software company estimates that avoidable forecast errors can add between 2% and 4% to costs of production.". Another example is "A survey (…) indicated that reductions in inventory levels resulting from improved forecast accuracy meant that a company with a $1 billion turnover could expect savings of between $5 million and $10 million" (Goodwin, 2018 p. 2).

Forecasting creates probable future scenarios and anticipates problems that can be considered long before happening. But, when a decision is being based on a forecast and, as time goes by, what was a "likely future scenario" changed dramatically. Or, in a more generic case, if the behavior of some metrics is not following a normal expected behavior? One example of this unexpected change of behavior is a very regular customer that suddenly stops buying. The possibility of generating warnings could alert that something is probably wrong and may require attention.

Another example is the alert generated by the high increase of purchases in a single product, that can anticipate the need for ordering it to the supplier. But, what seems to be something positive can be dramatically negative. For example, if the product price is wrong, and the product sold for a lower price than the purchase price. In this case, the organization is losing money, and it is fundamental to detect the problem as soon as possible.

This research results from cooperation between ISCTE University Institute of Lisbon and the company Primavera BSS. Primavera is a Portuguese technology company that

has established itself in the national market of computer management solutions responsible for the development of Enterprise Resource Planning (ERP) Software.

1.3.Research Methodology

We will use Design Science Research Methodology (DSRM) to develop this thesis, that focuses on "incorporates principles, practices, and procedures required to carry out such research" and meet three objectives: "consistent with prior literature", "nominal process model for doing DSR" and provides a mental model for presenting and evaluating DSR in IS " (Peffers, et al., 2008 p. 4). Artifacts are tools that enable us to solve the perceived problem, as displayed in Figure 1.

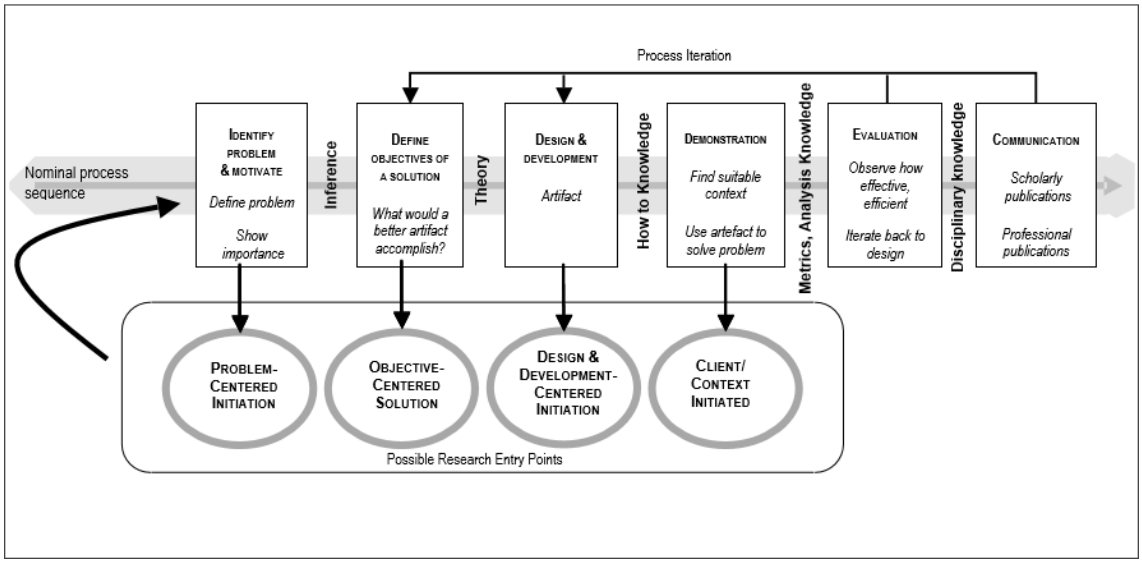


Figure 1- Design Science Research Methodology (Peffers, et al., 2008 p. 44)

We will use the Objective-Centered Solution research entry point since our goal is to learn the normal behavior of Business Metrics. We will create four artifacts: (1) A baseline instantiation for forecasting, (2) A proposed Prediction Model, (3) A proposed Anomaly Detection Model, and (4) A Comparative Performance Model to identify technique with better results. Then we will need to Demonstrate and Evaluate the artifacts ensuring they meet our research objectives (Figure 2).

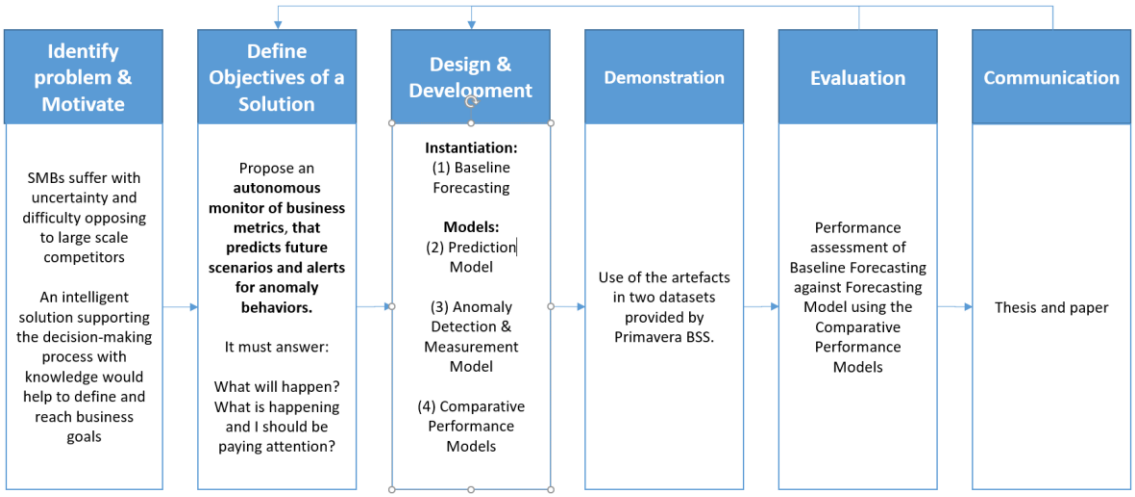


Figure 2 Guideline for DSRM (Adapted from Figure 1)

1.3.1. Identify Problem & Motivate

Small and medium-sized businesses (SMB) must deal with uncertainty and an increased difficulty opposing large scale companies that use data science to perform better. Decision Marking is typically supported by intuition and aggregated information like dashboards, that hides issues and opportunities. By incorporating in the Enterprise Resource Planning (ERP) a solution that uses Machine Learning to extract knowledge from Business Metrics, we are supporting the Data-Driven Decision Making with critical knowledge and helping define and follow Business Metrics Goals. In this way we are allowing to anticipate the identification of events (problems or opportunities) that can influence the performance of a company and help to achieve business goals.

1.3.2. Define the Objectives of the Solution

The objective is to develop an Autonomous Business Metrics Monitor supported by Machine Learning that will (1) Learn the normal behavior of business metrics; (2) Predict future behavior; (3) Identify events that represent an abnormal behavior; (4) score the impact of this events; (5) Alert for major incidents.

This solution must answer two questions: (1) What will happen? (2) What is happening, and someone should be paying attention?

1.3.3. Design & Development

To archive this goal, we are creating three artifacts

1. **A baseline for forecasting** (Instantiation) for evaluating one forecasting model developed by Primavera.
2. **A proposed Prediction Model**. Our goal is to develop a more accurate prediction model, that should perform forecasts in large time-series datasets.
3. **A proposed Anomaly Detection Model.** We intend to detect and measure deviations from expected behavior (anomalies), by using the predictive interval generated by the Prediction Model.
4. **A Comparative Performance** (model). This model will be used to measure the forecasting error of our proposed model. For the proposed anomaly detection model, we will use a synthetic time series dataset with anomalies and determine how well we can detect and score anomalies.

1.3.4. Demonstration

The relevance of developed artifacts is going to be demonstrated in two datasets provided by Primavera BSS. One dataset consists of, approximately 6.000 time series, with the monthly value of the Net Profit from approximately 6.000 different companies. Net Profit is the "amount of money the business makes after deducting the cost of sales and expenses from income" (klipfolio). A second dataset with synthetic time series was also generated by Primavera to demonstrate the Anomaly Detection Model.

1.3.5. Evaluation

For comparing the performance of the forecast methods, we use statistics based on forecasting errors, namely the mean of the SMAPE (Symmetric Mean Absolute Percentage Error) calculated between the forecasted and the observed values in the Net Profit dataset. SMAPE and other techniques for forecast accuracy measurement will be presented in Section 2.6. A hold out method will be applied to the data to remove one month, three months and six months. We will be creating three different train dataset and three different test datasets. The test dataset will be provided to the prediction artifact that will generate the forecast and prediction interval for each forecasting method that we will

test. At last, we will calculate the forecast error and compare the results between the baseline forecast and the proposed prediction model.

1.3.6. Communication

This research will be communicated by this document, submitted as an article to an intern technical journal of Primavera BSS and a paper. The paper will be submitted to an international conference with peer review.

1.4.Research Hypotheses

We will try to answer the following research questions:

RQ1 What are the most common Machine Learning techniques for Autonomous Forecast of Business Metrics?

RQ2 What is the performance improvement of the proposed Autonomous Forecast Model versus a baseline?

RQ3 Can we detected deviations from expected behavior (anomalies) by using the prediction interval of the forecast model?

1.5.Document Structure

The present study is organized in five chapters that intend to reflect the different phases of the research methodology until its conclusion.

The first chapter introduces the research theme, objectives, methodology, as well as a brief description of the work structure.

The second chapter reflects the theoretical framework, called the Related Work.

The third chapter presents the cases study used in our study and the design and development details of forecast and anomaly detection.

The fourth chapter presents the analysis of the results obtained according to the appropriate methodology.

The fifth and final chapter presents the conclusions of this study as well as the recommendations, limitations, and future work.

2 – Related Work

This chapter of the dissertation addresses the theoretical foundations and work related to the problem already identified, in the elaboration of predictions at scale, as well as in the detection of anomalies. The content of this chapter will serve as the basis for the following chapters.

This chapter is organized in the following Sections:

- Section 2.1 – Introduces Data Science and its importance in business application supporting the decision-making process. We present four types of Decision-Making Models: "Human Judgment," "Data-Driven," "AI-Driven" and "Combine AI and Human Judgment";
- Section 2.2 –Highlights the principal characteristics of Time Series and Business Time Series;
- Section 2.3 – Shows the importance of extracting Time Series Features and Visualization;
- Section 2.4 – Introduces the most used predictive modeling techniques in business time series;
- Section 2.5 – Present fitted values and residuals resulting from different time series models;
- Section 2.6 – Shows how to measure and evaluate the forecasting accuracy
- Section 2.7 – Presents prediction intervals that represent the uncertainty in the forecasts
- Section 2.8 – Introduces the Meta-Learning Forecast where a trained classifier will choose the best predictive model
- Section 2.9 – Introduces anomaly detection

2.1. Data Science and its importance in business application

In (Provost, et al., 2013 p. 2), Data Science is defined as "a set of fundamental principles that guide the extraction of knowledge from data" which differs from the definition of Data Mining which represents "the extraction of knowledge through technologies" (Provost, et al., 2013 p. 2) . The same authors present a collection of the most important fundamental concepts of data science that includes the process from problem definition, the application of data science techniques, and the implementation of results to improve decision making:

Fundamental Concepts of Data Science Applied to Business:

1- How data science fits into an organization, and ways in which the Data Science team must organize and access data to create a competitive advantage.

2- How to approach the data analytically to identify methods that are considered appropriate.

3- Ways to extract knowledge from data that support a wide range of data science tasks and their algorithms.

It is important to clarify the difference between Data Mining, Machine Learning, and Artificial Intelligence (AI). In (Gartner) Glossary - Information Technology, Data Mining is defined by the "process of discovering meaningful correlations, patterns, and trends by sifting through large amounts of data." Machine Learning is defined as "composed of many technologies (such as deep learning, neural networks, and natural-language processing), used in unsupervised and supervised learning, that operate guided by lessons from existing information." AI "applies advanced analysis and logic-based techniques, including machine learning, to interpret events, support and automate decisions, and take actions."

A direct relationship between data-based decision making and company performance is demonstrated in (Brynjolfsson, et al., 2011). Results suggest that Data-Driven Decision "can be modeled as intangible assets which are valued by investors and which increase output and profitability." (Brynjolfsson, et al., 2011)

In (Colson, 2019) four types of Decision-Making Model are compared: "Human Judgment," "Data-Driven," "AI-Driven," and "Combine AI and Human Judgment."

- In **Human Judgment** Professional "relied on their highly tuned intuitions, developed from years of experience (and a relatively tiny bit of data) in their domain";
- In **Data-Driven Decision Making** "Human judgment is still the central processor, but now it uses summarized data as a new input";
- In **AI-Driven** "bring AI into the workflow as a primary processor of data";
- Finally, in **Combine AI and Human Judgment,** it is proposed to "Leveraging both AI and Human processors in the workflow."

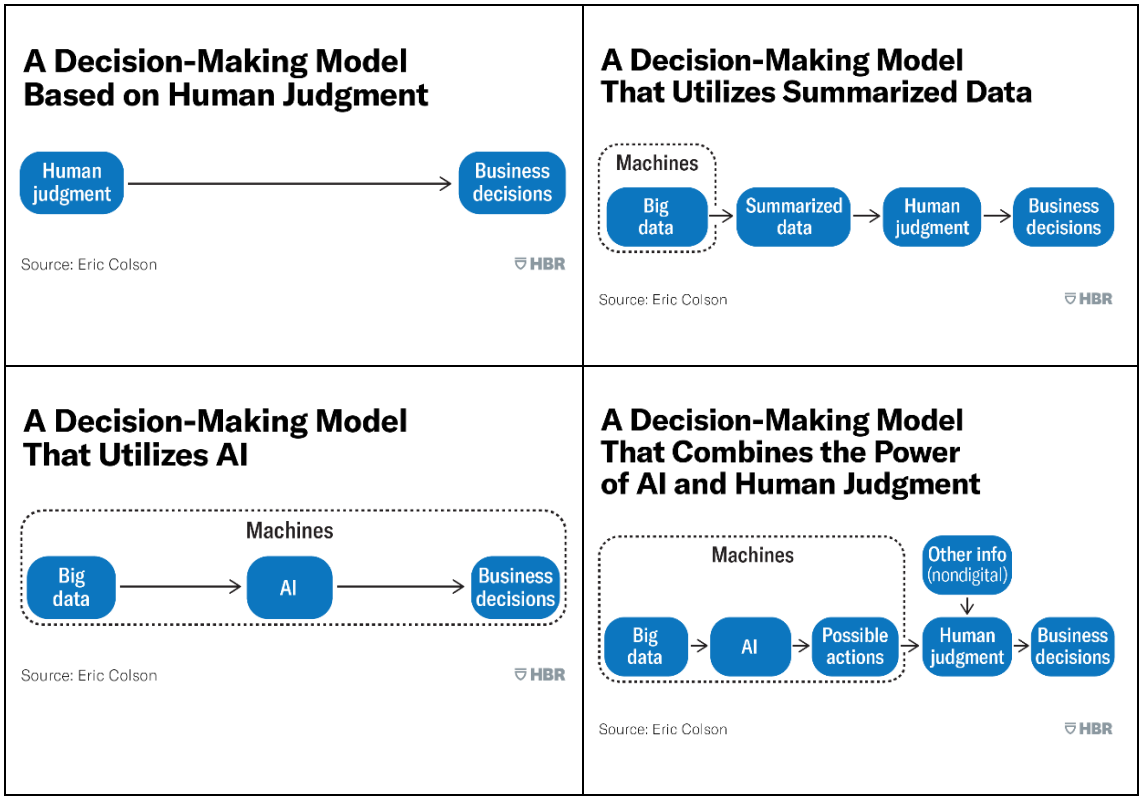


Figure 3 – Four type of Decision-Making Models (Colson, 2019)

Knowledge extracted from an organization's data has other uses than the ability to support decisions. In (Provost, et al., 2013 p. 9) it is presented that “the ability to extract useful knowledge from data should be considered as a strategic asset,” thus allowing to evaluate the possibility of making investments. This increase in the ability to extract knowledge from data to strategic assets is presented as one of the fundamental principles of data science. Increasing revenue and lowering costs is achieved through the results of data science teams working to exploit data in traditional businesses.

2.2. Characteristics of Time Series and Business Time Series

A time series is defined as "a sequence of data points, typically measured at uniform time intervals" (discrete time series) that can be broken down into four components: trend. seasonal, cyclic, irregular (Hyndman, et al., 2019) Section 2.3

- Trend: This component becomes evident when there is a long-term increase or decrease in the data. Also referred to as a "change of direction," when it can, for example, change from an upward trend to a downward trend.
- Seasonal: A seasonal pattern in a time series occurs when affected by periodic factors, such as the day of the week or the time of the year. It is a fixed and known frequency.
- Cyclic: A cycle occurs when data displays non-fixed frequency ups and downs. These changes are usually due to economic conditions and are often related to the "business cycle." The duration of these fluctuations is usually at least two years, but in general, it is not constant like in the seasonal component.
- Irregular is a component that contemplates what is not explained by the other components.

The same author clarifies the difference between cyclical behavior and seasonal behavior: If changes are not fixed frequency, then they are cyclical, otherwise if frequency is associated with some aspect of the calendar, the default is to be seasonal. Thus, the regularity (or its absence) makes the difference between seasonal and cyclical components.

It is also concluded that, in the case the cyclical and seasonal effects are not relevant for the series, the forecast of future values will necessarily be based on the components trend and irregular, however, as the irregular component is difficult to model and predict, the forecasting process can only be supported by the trend.

Business Time Series

In (Taylor, et al., 2017 pp. 4,5) the authors identify the existence of a "wide diversity of business forecasting problems " and a set of common characteristics typically present in this type of time series are:

- **Seasonality in Business Time Series:** The seasonal effect typically has a set of overlapping cycles. Thus, there can be a weekly cycle and an annual cycle. Additionally, there are typically evident effects in festive seasons such as Christmas and New Year.
- **Trend in Business Time Series:** Changes in trend may arise, such as the impact of market changes. This type of trend change was previously identified as the Cyclical component.
- **Outliers:** As with any real database, a set of Outliers is also expected. The authors use the example of the number of events created on Facebook to highlight the features previously presented and to suggest that this type of time series highlights "the difficulties of producing reasonable forecasts with fully automated forecasting methods."

2.3. Time Series Features and Visualization

In the talk Feature-based time series analysis (Hyndman, 2018), introduces the necessity to use time-series features to analyze and extract knowledge from a hight amount of time series dataset. Time series feature can be described as “any measurable characteristic of a time series” (Talagala, et al., 2019 p. 4).

Global features is referred to characteristics that “quantify patterns in time series across the full-time interval of measurements” and can “distill complicated temporal patterns” into “interpretable low-dimensional summaries” (Fulcher, 2017 p. 8).

It is defined that global features can be applied to time series of variable lengths and “allows a time-series dataset to be represented as a time series matrix” (Fulcher, 2017 p. 9) . In Figure 4, it can be seen as an example where a time series dataset is converted into a time series features matrix. It is suggested that this matrix can be used to enable a range of analysis tasks and a low-dimensional structure, or discriminative features for time-series classification.

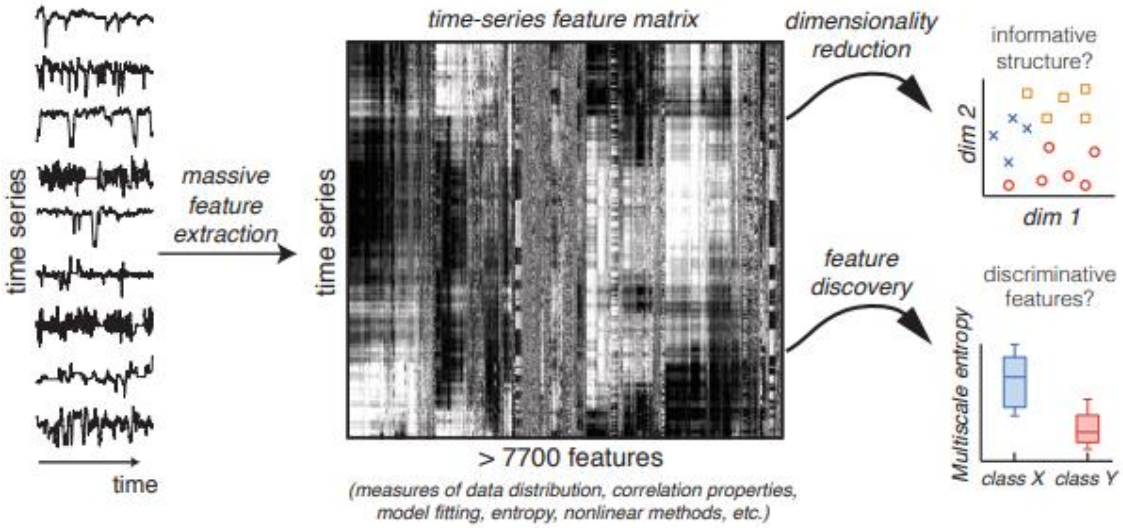


Figure 4 Time series dataset (left) converted to a time series features matrix. (Fulcher, 2017 p. 14)

Some examples of definitions of time series features are presented in (Talagala, et al., 2019 pp. 23 - 25):

- **Length of time series** – how many observations are available
- **Features based on an STL-decomposition** – "The strength of trend, strength of seasonality, linearity, curvature, spikiness and first autocorrelation coefficient of the remainder series, are calculated based on a decomposition of the time series."
- **Stability and lumpiness** "are calculated based on tiled windows (i.e., they do not overlap)."
- **Spectral entropy of a time series** "is based on information theory, and can be used as a measure of the forecastability of a time series."
- **Hurst exponent** that "measures the long-term memory of a time series."
- **Nonlinearity** that "measures the degree of nonlinearity of the time series."

2.4. Predictive Modeling in Business Time Series

In (Hahmann, 2019 p. 180) Time series forecasting (or prediction as it explicit referred as a synonym) is defined as "the process of making predictions of the future values of a time series, i.e., a series of data points ordered by time, based on its past and present data/behavior". Additionally, it is mentioned the importance of forecasting for planning and decision making.

In (Goodwin, 2018 p. 2) it defined that, in business, "forecasting demand for products and services may be crucial to the operations of most companies." For example, processes such as inventory, cash flow or logistics planning, production, human resource decisions or purchasing may depend on forecasts.

The same authors identify that good predictive performance "will result in better levels of customer service and thus promote customer satisfaction and retention." For example, by mitigating costly emergency production cycles and reduced waste (and investment) associated with excessive stock levels.

In (Ord, et al., 2017 p. xiii) it is mention that "virtually every manager has to make plans or decisions that depend on forecasts" but recognized that "there can never be just one approach to forecasting that meets all needs, rather we must invest in horses for courses[1]" proposing a forecasting task composed by 6 elements "Purpose, Information, Value, Analysis, System and Evaluation" suggesting to summarize by the mnemonic PIVASE

- Purpose – "What do we hope to achieve by generating the forecast? That is, what plans are dependent upon the results of the forecasting exercise? How far ahead do we wish to forecast? We refer to this period as the forecasting horizon."
- Information – "What do we know that may help us in forecasting. And when will we know it? Detailed data is only useful if it is available in a timely fashion."
- Value – "How valuable is the forecast? What would you pay to have perfect knowledge of the future event?"

[1] "Horses for courses" It's a British proverb that means that "Different people are suited to different things"

- Analysis, “From analyzing the data, can we develop a model that captures its characteristics? And how does it perform on new (hold-out sample) data?”
- System. “What system of forecasts (models) and software is needed to meet the needs of the organization?”
- Evaluation “How do we know whether a particular forecasting exercise was effective, and what the potential is for improvement?”

Same authors suggest some key principles for a good forecasting practice (Armstrong, 2001)

1. “**Ensure the data match the forecasting situation,** (…) examine the available data with respect to the end-use (…) and make sure a match exists”.
2. “**Clean the data**, (…) as data may be omitted, wrongly recorded, or affected by changing definitions.”
3. “**Use transformations as required by the nature of the data,** examined differences, growth rates, and log transforms.”
4. “**Use graphical representation of the data.** Highlight key events, plotting the data can provide a variety of insights.”
5. “**Adjust for unsystematic past events (E.g., Outliers)”** as some factors like: “weather, political, supply sabotages,” “need to be considered when clear reasons can be identified for the unusual observations.”
6. “**Adjust for systematic events (E.g., Seasonal effects)**” like “weekends, public holidays, seasonal patterns…”
7. “**Use error measure that adjust for scale in the date when comparing across series**” using MAE or RMSE for a single Series or scale-free MAPE (if appropriated) of relative errors like MAE or MASE
8. “**Use multiple measures of performance**-based upon the observed forecasting errors” (…) using multiple measures allowing users to use most relevant for their needs.

We will introduce in the next pages a set of time series forecasting techniques.

2.4.1 Simple forecasting Methods

A set of extremely simple and surprisingly effective forecasting methods are presented in Table 1 (Hyndman, et al., 2019) Section 3.1

Method Name	Forecast behavior
Average	The average of the historical data
Naïve	The value of the last observation
Seasonal naïve	The last observed value from the same season of the last year
Drift	It is like the naïve method but allowing the forecasts to increase or decrease by a Drift amount of change (the average change seen in the historical data)

Table 1 Simple time series forecasting methods, based in (Hyndman, et al., 2019) Section 3.1

Next, we will introduce to Exponential Smoothing and ARIMA methods, described by the same authors as "the two most widely used approaches to time series forecasting," that provide "complementary approaches." Exponential Smoothing is briefly described as "based on a description of the trend and seasonality in the data," The ARIMA models "aim to describe the autocorrelations in the data." (Hyndman, et al., 2019) Chapter 8

2.4.2 Forecasting with Exponential smoothing

Exponential smoothing method was proposed in 1950, and "motivated some of the most successful forecasting methods" (Hyndman, et al., 2019) Chapter 7, defined here by "a weighted average of past observations, with the weights decreasing exponentially as observations get older." Also, the importance for the industry application if referred by generating "reliable forecasts quickly and for a wide range of time series."

Simple exponential smoothing can be express by:

$$y_{T+1|T} = \alpha y_T + \alpha(1-\alpha)y_{T-1} + \alpha(1-\alpha)^2 y_{T-2} + \cdots,$$

The T + 1 is a weighted average of all the observations in the series $y_1,\ldots,y_T$ and $0 \leq \alpha \leq 1$ is the smoothing parameter that controls the rate which the weights decrease.

In Table 2, the various exponential smoothing methods are presented.

Exponential Smoothing types	Description
Simple	Time Series without a clear trend or seasonal pattern
Double (Holt model)	Extension of Simple Expo. Smoothing that adds trend support.
Triple (or Holt-Winters)	Extension of Double Expo. Smoothing that adds seasonality support

Table 2 Exponential smoothing methods based in (Brownlee, 2018)

Finally, to warrant that seasonality is modeled correctly, the number of steps in a seasonal period needs to be specified. For example, in a monthly data time series, the seasonal period is typically twelve if the behavior tends to repeat each month.

2.4.3 Forecasting with ARIMA

The word ARIMA is an acronym for Autoregressive Integrated Moving Average. (Hyndman, et al., 2019) Chapter 8 suggests, to understand the ARIMA model, it is first necessary to understand the concept of stationarity and differentiation.

In a stationary time series, properties do not depend on the time it is observed. This means that a time series with trend or seasonality cannot be considered stationary because "trend and seasonality will affect the value of time series at different times." On the other hand, a series of white noise is stationary – it will look identical, no matter when it is observed.

One away of making a stationary time series from a non-stationary one, called differencing, is by computing the difference between consecutive observation:

$$y'_t = y_t - y_{t-1}$$

Where y_t is one observation and $y_{t-1}$is the previous observation. The difference will have only T-1 values since it is not possible to calculate a difference for the first observation.

Occasionally it may be necessary to differentiate a second time to obtain a stationary time series, called a second-order differencing. This process can help stabilize the mean of a time series by reducing (or eliminating) trend and seasonality. In the ARIMA

acronym presented before (AutoRegressive Integrated Moving Average), the "integrated" is the inverse of differentiation.

$$\begin{aligned} y''_t &= y'_t - y'_{t-1} \\ &= (y_t - y_{t-1}) - (y_{t-1} - y_{t-2}) \\ &= y_t - 2y_{t-1} + y_{t-2}. \end{aligned}$$

To explain Auto Regression, these authors compare with a multiple regression model where it is used as a linear combination of predictors for forecasting a variable of interest. As the term autoregression indicates, in this model, the forecast of the variable uses a combination of past values of itself, as can be written as

$$y_t = c + \phi_1 y_{t-1} + \phi_2 y_{t-2} + \cdots + \phi_p y_{t-p} + \varepsilon_t$$

Where c and ϕ_n are respectively the constant term and the slope coefficients in the regression model, and ε_t is a white noise

Finally, the Moving Average Models uses past forecasting errors in a regression-like model rather than using the past value of the variable.

$$y_t = c + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q}$$

Where c and ε_t are respectively the constant term and the slope coefficients in the regression model, and ε_t is a white noise

Non-Seasonal and Seasonal ARIMA Model can be obtained by combine differencing with autoregression and moving average

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

Where c and ϕ_n are respectively the constant term and the slope coefficients in the regression model, ε_t is a white noise, and y'_t is the differenced series

Authors called this model as a $\mathrm{ARIMA}(p, d, q)$ where:

'p' is the order of the autoregressive component;

'd' the degree of differencing (like first-order, second-order differencing);

'q' the order of the moving average component.

Autor refers some special examples of ARIMA models

White noise	ARIMA(0,0,0)
Random walk	ARIMA(0,1,0) with no constant
Random walk with drift	ARIMA(0,1,0) with a constant
Autoregression	ARIMA(p,0,0)
Moving average	ARIMA(0,0,q)

Table 3 Some special cases of Arima Models (Hyndman & Athanasopoulos, 2019) Section 8.5

A Seasonal ARIMA model is formed by assigning a seasonal term in the non-seasonal Arima Model $\mathrm{ARIMA(p, d, q)}\ (P, D, Q)m$, where m is the number of seasons per year. It is used uppercase for seasonal parts and lowercase for non-seasonal parts of the model.

2.4.4 Forecasting with Theta

The Theta method is described as a version of Exponential Smoothing, having created academic interest for the performance achieved in the M3 competition (Hyndman, et al., 2001 p. 1). The M competition aims to evaluate and compare the accuracy of different forecasting methods (MCompetitions). The first edition was in 1982, followed by three editions in 1993, 2000, and 2019. For the M3 competition (the year 2000), 3003 time series from different areas such as industry, finance, and demographics were available, mostly with Monthly time intervals, Quarterly and Annual. Twenty-four methods were tested where the Theta method had the best results in virtually all data types.

In the next point, we will introduce Prophet, a forecasting at scale model developed by Facebook Research.

2.4.5 Forecasting with Prophet

The Prophet prediction model is presented in (Taylor, et al., 2017 p. 1) as "a modular regression model with interpretable parameters that can be intuitively adjusted by analysts with domain knowledge about the time series ", and it is described "performance analyses to compare and evaluate forecasting procedures, and automatically flag forecasts for manual review and adjustment"

In (Taylor, et al., 2017_2) the Prophet Model is presented as being developed by Facebook Research and made open-source in February 2017. It is indicated that "has been a critical piece in enhancing Facebook's ability to create a large number of reliable predictions used for decision making and even product functionality." An given example of the application of prophet is the capacity planning in order to allocate resources and define goals to measure performance. Additionally, the authors refer that the Prophet Prediction Model is implemented on the R and Python platforms, Open Source, and "is a prediction model designed to deal with business time series characteristics and used decomposable time series model."

$$y(t) = g(t) + s(t) + h(t) + \varepsilon$$

on:

- g(t) trend, models non-periodic changes
- (t) periodic monthly / annual seasonality
- h(t) effect of holidays occurring on potentially irregular markings on one or more days
- ε error – idiosyncratic changes that are not explicable by the model. Error is assumed to follow a normal distribution.

An advisable approach to the production of scaled forecasts, called Analyst-in-the-loop, has been defined. The purpose is to make the best possible use of human and automated tasks.

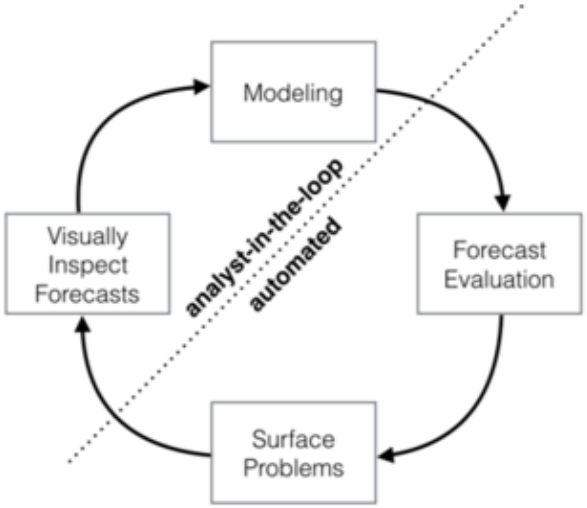


Figure 5 Analyst-in-the-loop approach to make the best possible use of human as well as automated tasks (Taylor, et al., 2017 p. 3)

Analyst-in-Loop Approach:

1- Model Time Series with flexible specification (interpretable)
2- Produce forecasts for this baseline set model for a variety of forecast simulation dates on historical dates that allow you to evaluate performance.
3- Flag human analyst when there is a poor performance
4- The analyst may adjust the inspection-based model to the forecast

2.5. Fitted values and Residuals

The fitted values can be obtained from a forecasting model. The fitted value of each observation is the one-step forecast using all previous observations.

Fitted values are denoted by: $\hat{y}_{(t|t-1)}$ meaning the forecast of y_t based on observations $y_1, \ldots, y_{t-1}$ (Hyndman, et al., 2019) Section 3.3.

Authors correct that fitted values are not true forecasts. The reason is that "any parameters involved in the forecasting method are estimated using all available observations in the time series, including future observations."

The residuals are "what is left over after fitting a model." Those values are useful to check if the "model has adequately captured the information in the data," or in other words, it is "using all of the available information." In this case, the residuals should be uncorrelated, and a zero mean. If the mean is different than zero, we have a biased forecast that can be improved by simply adding the mean value to all forecasts.

2.6. Evaluate Forecasting Accuracy

The accuracy of a forecasting procedure is considered a key question, and the "E" in PIVASE mnemonic suggested by (Ord, et al., 2017 p. 43) that was introduced earlier.

For these authors, the evaluation allows to compare different forecasting procedure and to select the one with the best record. On the other hand, when the forecasting method is being used regularly, we need to evaluate its accuracy to verify if the expected performance remains. The suggest procedure is "look at the differences between the observed value and the forecasts, and to use their average as a performance measure" (Ord, et al., 2017 p. 45)

In (Hyndman, et al., 2019) Section 3.4, alerts that, to measure the accuracy, we must evaluate "how well a model performs on new data that were not used when fitting the model." We need to separate the available data in two portions: Training used for fitting the model, and the test data, used to evaluate its accuracy. This procedure gives us a vision of how well the model will perform when forecasting on new data.

It is said that the size of the test portion, also describe by some references as "hold-out set," depending on how long the sample is, but that typically is 20% of the total sample, or "at least as large as the maximum forecast horizon required." (Hyndman, et al., 2019) Section 3.4

Same authors alert for importance of evaluating the forecasting accuracy and that fitting the train data well does not necessarily mean that a model will forecast well. It is possible to obtain a perfect fit (in the train data) with enough model parameters, but this over-fitting is "just as bad as failing to identify a systematic pattern in the data." (Hyndman, et al., 2019) Section 3.4

When evaluating the forecasting accuracy, it is measured the forecast error. Same authors notice that this error "does not mean a mistake; it means the unpredictable part of an observation."

$$e_{T+h} = y_{T+h} - \hat{y}_{(T+h|T)}$$

training data is given by $\{\boldsymbol{y_1}, \dots, \boldsymbol{y_T}\}$ and the test data is given by $\{\boldsymbol{y_{T+1}}, \boldsymbol{y_{(T+2)}}, \dots\}$

Authors clarify the difference between the forecasting error that is calculated on the test set and can involve multi-step forecast, and the residual that is calculated in the training set and based on the one-step forecast.

The Absolute Error (AE) is defined in (Ord, et al., 2017 p. 47) as "the value of error regardless of its sign":

Absolute Error (AE) = $|e_i| = |Y_i - \hat{Y}_i|$

According to (Hyndman, et al., 2019) Section 3.4, forecast accuracy can be measured by scale-dependent, percentage, or scaled errors.

It is defined the forecast "error" as "difference between an observed value and its forecast" It can be written as

$$e_{T+h} = y_{T+h} - \hat{y}_{(T+h|T)}$$

where the training data is given by $\{\mathbf{y_1}, \ldots, \mathbf{y_T}\}$ and the test data by $\{y_{T+1}, y_{T+2}, \ldots\}$

Scale-Dependent Errors

Scale-dependent errors "cannot be used to make comparisons between series that involve different units.". (Hyndman, et al., 2019) Section 3.4

The "most commonly" Scale-dependent errors used are:

Mean absolute error (MAE) = $mean(|e_t|)$,

Root mean square error (RMSE) = $\boxed{\sqrt{\text{mean}(e_t^2)}}$

Authors identify that MAE is easier to compute and understand as it is a simple mean of the, so called "forecast error".

Percentage errors

Percentage errors have "the disadvantage of being infinitive/undefined if $y_t = 0$ for any *t*" because it will be divided by zero (see the equation below). On the other hand, it

is not symmetric as it puts "a heavier penalty on negative errors than on positive errors" (Hyndman, et al., 2019) Section 3.4

The "most commonly used" is MAPE:

Mean absolute percentage error: MAPE = $mean(|100e_t\ /y_t\ |)$

In (Hyndman, 2014) the sMAPE (Sysmetric MAPE is defined as:

Symmetric MAPE (sMAPE) = 100 $mean(2|y_t - \widehat{y_t}\ |/(y_t + \widehat{y_t}))$

Same author alerts for the different definitions of SMAPE in the literature. It was argued that MAPE "puts a heavier penalty on forecasts that exceed the actual than those that are less than the actual" and SMAPE tries to eliminate this penalty.

Scaled Errors

Scaled errors are "an alternative to using percentage errors when comparing forecasts" and "scaling the errors based on the training MAE from a simple forecast method."

For non-seasonal time series, it is suggested to define a scaled error that uses naïve forecast:

$$q_j = \frac{ej}{\frac{1}{T-1}\sum_{t=2}^{T}|y_t - y_{t-1}|}$$

For seasonal time series, it is suggested to define a scaled error that uses seasonal naïve forecasts

$$q_j = \frac{ej}{\frac{1}{T-m}\sum_{t=m+1}^{T}|y_t - y_{t-m}|}$$

The mean absolute error is defined as

Mean Absolute Error (MASE) = $mean(|qj|)$

2.7. Prediction Intervals

A prediction interval is defined as an interval "within which we expect y_t to lie with a specified probability", expressing in this way the "uncertainty in the forecasts." (Hyndman, et al., 2019) Section 3.5

Prediction interval for the h-step forecast, assuming the forecast errors are normally distributed, and a 95% prediction interval is:

$$\hat{y}_{(T\ +\ h|T)} \pm 1.96\hat{\sigma}_h$$

The multiplier (1.96) depends on the confidence level. The table below shows some multipliers (resulting from the standardized normal distribution) that can be used for prediction intervals:

Confidence level	Multiplier
50	0.67
55	0.76
60	0.84
65	0.93
70	1.04
75	1.15
80	1.28
85	1.44
90	1.64
95	1.96
96	2.05
97	2.17
98	2.33
99	2.58

Table 4 Some multipliers that can be used to calculate the prediction intervals (Hyndman, et al., 2019) Section 3.5

Prediction Intervals can be "easier calculated" if the residuals have constant variance and are normally distributed. A "Box-Cox transformation may assist" if a forecasting method does not satisfy these two properties. (Hyndman, et al., 2019) Section 3.5

In (Ord, et al., 2017 p. 51) an Empirical Prediction Intervals is presented when we cannot assume that predictive distribution follows the normal law.

2.8. Meta-Learning Forecasting

FFORMS (Feature-based FORecast-model Selection), is a "general framework for forecast-model selection using meta-learning" (Talagala, et al., 2019 p. 2). In this paper, it is classified as challenging the selection of the most appropriated model for forecasting. It is acknowledged the two most commonly used automated forecast methods: Exponential smoothing (ets) and ARIMA (auto-arima). Both algorithms work as a class of models where the best model is automatically chosen. But the appropriated class of models relies on the "expert judgment of the forecaster" and if there is sufficient amount of data a hold-out test can be made to choose the best class of models, if there isn't sufficient amount of data, it is suggested a cross-validation that "increases the computation time involved considerably" (Talagala, et al., 2019 p. 3). The need of a "fast and scalable algorithm to automate the process of selecting models" motivates the study and proposes the FFORMs Framework presented.

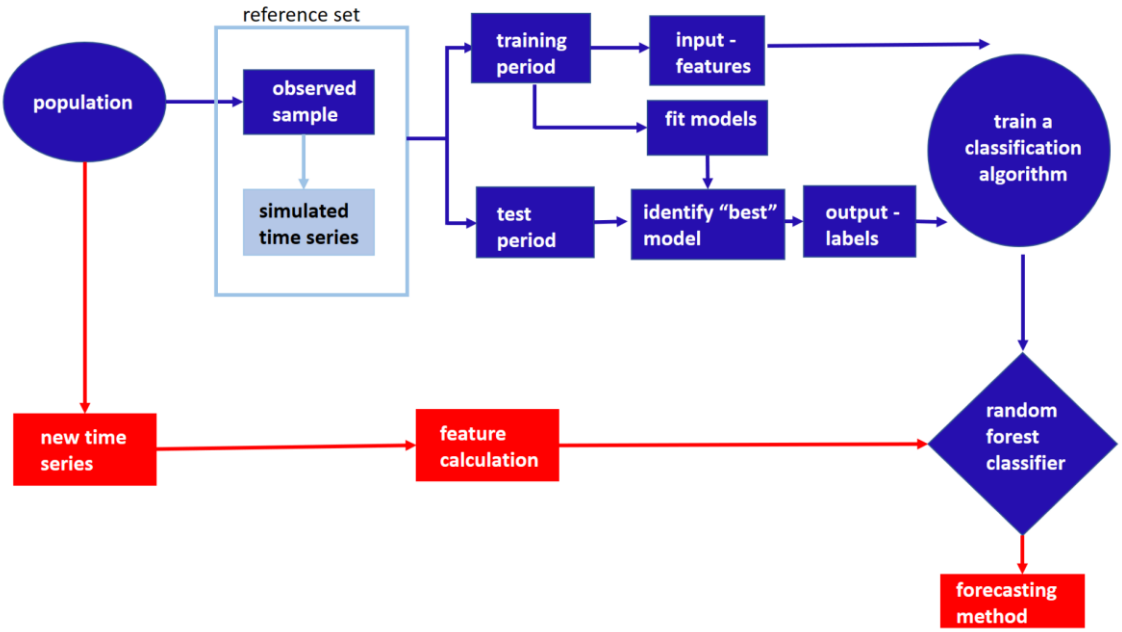


Figure 6 - FFORMs Framework (Talagala, et al., 2019 p. 10)

The framework is divided into two phases, an "offline phase" that is shown in blue and an "online phase" that is shown in red. In the offline phase, the "classifier is built using a large historical collection of time series, in advance of the forecasting task at hand." In

the online phase "involves calculating the features of a time series and using the pre-trained classifier to identify the best forecasting model." The key element in this proposed framework is to work with time-series features (any measurable characteristic) rather than work directly with individual observations of the time series.

2.9.Anomaly Detection

In (Chandola, et al., 2009 p. 1) Anomaly detection refers to the problem of "finding patterns in data that do not conform to expected behavior." It is pointed that Anomaly detection is often referred, in different domains, as anomalies, outliers, discordant observations, exceptions, aberrations, surprises, peculiarities, or contaminants.

(Shipmon, et al., 2017 p. 1) referrers that at Google, time series anomaly detection is used for the detection of unexpected drops in traffic. That may be an early warning of an issue, and that potentially remedial action may be necessary. The need to define a rule to identify anomalies by comparing the prediction can be done by calculating the Euclidean distance and setting a threshold. To mitigate high false positives and detect continuous outages they suggest the using of an accumulator, that "increment the counter for every local outage and decrement it for every non-anomalous value" (Shipmon, et al., 2017 p. 2) , and the use of a probabilistic approach that compares short-term and long-term variance.

In (Chandola, et al., 2009) the types of anomaly detection techniques are grouped in Classification Based, Nearest Neighbor Based, Clustering Based, Statistical, Information Theoretic and Spectral

Classification Based Anomaly Detection Techniques: "used to learn a model (classifier) from a set of labeled data instances (training) and then, classify a test instance into one of the classes using the learnt model (testing)" (Chandola, et al., 2009 pp. 20,21) Some techniques that uses different classification algorithms are can be Neural Network Based, Bayesian Networks Based, Support Vector Machines Based and Rule Based.

Nearest Neighbor Based Anomaly Detection Techniques "require a distance or similarity measure defined between two data instances" and are based on the assumption that "Normal data instances occur in dense neighborhoods, while anomalies occur far from their closest neighbors". (Chandola, et al., 2009 p. 25)

Clustering Based Anomaly Detection Techniques "is used to group similar data instances into clusters. Clustering is primarily an unsupervised technique though semi-supervised clustering" (Chandola, et al., 2009 p. 30). It is suggested three different categories based on different assumptions: (1) "Normal data instances belong to a cluster

in the data, while anomalies either do not belong to any cluster." (2) "Normal data instances lie close to their closest cluster centroid, while anomalies are far away from their closest cluster centroid.", and (3) "Normal data instances belong to large and dense clusters, while anomalies either belong to small or sparse clusters." (Chandola, et al., 2009 pp. 30,31)

Statistical Anomaly Detection Techniques "fits statistical model (usually for normal behavior) to the given data and then apply a statistical inference test to determine if an unseen instance belongs to this model or not" and are based on the assumption that: "Normal data instances occur in high probability regions of a stochastic model, while anomalies occur in the low probability regions of the stochastic model."

Information Theoretic Anomaly Detection Techniques "analyze the information content of a data set using different information theoretic measures such as Kolomogorov Complexity, entropy, relative entropy, etc" and are based on the assumption that: "Anomalies in data induce irregularities in the information content of the data set."

Spectral Anomaly Detection Techniques "try to find an approximation of the data using a combination of attributes that capture the bulk of variability in the data" and are based on the assumption that "Data can be embedded into a lower dimensional subspace in which normal instances and anomalies appear significantly different."

In (Sakr, et al., 2019) Anomaly Detection is presented in different Areas:

Area of Study	Synonyms	A.D. appear as
Big Data in Network **Anomaly Detection**	Network anomaly detection; Network normal traffic/ behavior modeling; Network outlier detection	"Network **anomaly detection** refers to the problem of finding anomalous patterns in network activities and behaviors, which deviate from normal network operational patterns. More specifically, in network anomaly detection context, a set of network actions, behaviors, or observations is pronounced anomalous when it does not conform by some measures to a model of profiled network behaviors, which is mostly based on modeling benign network traffic." Examples of anomalies on the network anomalies are mostly categorized into two types: **performance-related anomalies** and **security-related anomalies** (Sakr, et al., 2019 pp. 283, 284)
Business Process Anomaly Detection	Business Process Deviance Mining	"Business process deviance mining refers to the problem of (automatically) detecting and explaining deviant executions of a business process based on the historical data stored in a given Business Process Event Log (called hereinafter event log for the sake of conciseness). In this context, a deviant execution (or "deviance") is one that deviates from the normal/desirable behavior of the process in terms of performed activities, performance measures, outcomes, or security/compliance aspects." (Sakr, et al., 2019 p. 389)
Big Data in Computer Network Monitoring		"Techniques for **anomaly detection** that are often employed in cybersecurity tasks. For example, anomaly detection is used in network security contexts to detect traffic related to security flaws, virus, or malware, so to trigger actions to protect users and the network." (Sakr, et al., 2019 p. 262)
Big Data in Mobile Networks		**"Anomaly detection (AD)** is a discipline aiming at triggering notifications when unexpected events occur. The definition of what is unexpected – i.e., what differs from a normal or predictable behavior – strictly depends on the context. In the case of mobile networks, it might be related to a number of scenarios, for example, degradation of performance, outages, sudden changes in traffic patterns, and even evidences of malicious traffic. All these events can be observed by mining anomalous patterns in passive traces at a different level of granularities." (Sakr, et al., 2019 p. 279)
Pattern Recognition		Complex event recognition (CER) "has also been used to provide real-time access to monitoring data and enable anomaly detection and resource optimization in the context of grid infrastructures" (Sakr, et al., 2019 p. 1260)
Scalable Architectures for Big Data Analysis		"Graphs are immensely useful for data mining applications, such as social influence analysis, recommendations, clustering, and anomaly detection." (Sakr, et al., 2019 p. 1450)

Table 5 Different areas where anomaly detection is presented (Sakr & Albert, 2019)

In (Mehrotra, 2017 p. 4) Anomalies or Outliers (which are used synonymously by the authors) are referred to as "substantial variations from the norm." The detection of these anomalies is referred to as "based on models and predictions from past data."

The authors suggest the formulation of 5 questions for the formulation of anomaly detection algorithms: (Mehrotra, 2017 p. 6)

- "How is the norm characterized?"
- "What if there are multiple and substantially different cases, all of which can be considered to be normal?"
- "What can we consider as a substantial variation, as opposed to a minor variation from a norm?"
- "How do we address multi-attribute data?"
- "How do we address changes that happen over time?"

+

The paper (Zhu, et al., 2017) "motivates a real-world anomaly detection use-case at Uber (…) to improve performance at scale" (Zhu, et al., 2017 p. 2). The importance of joining an "accurate time series forecasting" with a "reliable estimation of the prediction uncertainty" is declared as "critical for anomaly detection, optimal resource allocation, budget planning, and other related tasks. "The prediction uncertainty is important for assessing how much to trust the forecast produced by the model and has a profound impact in anomaly detection" (Zhu, et al., 2017 p. 1)

These authors propose a "novel end-to-end model architecture for time series prediction, and quantify the prediction uncertainty using Bayesian Neural Network, which is further used for large-scale anomaly detection." (Zhu, et al., 2017 p. 1) They propose to decompose and quantify the prediction uncertainty from the sources:

Model uncertainty (or epistemic uncertainty) that "captures our ignorance of the model parameters and can be reduced as more samples being collected."

Inherent noise "captures the uncertainty in the data generation process and is irreducible."

Model misspecification "captures the scenario where the testing samples come from a different population than the training set, which is often the case in time series anomaly detection." (Zhu, et al., 2017 p. 1)

An example of the utility of the prediction interval in anomaly detection is provided as “alerts will be fired when the observed value falls outside the constructed 95% interval”. (Zhu, et al., 2017 p. 3)

According to (Chandola, et al., 2009 p. 7) it is specified three categories of anomalies:

Point Anomalies “If an individual data instance can be considered as anomalous concerning the rest of data, then the instance is termed as point anomaly.” In this case, it’s not considered any contextual or behavioral attribute and typically used in simple outlier detection scenarios.

Contextual Anomalies “If a data instance is anomalous in a specific context, then it is termed as a contextual anomaly and also referred to as a conditional anomaly. Each data instances are classified into either contextual or behavioral attribute

Collective Anomalies “If a collection of related data instances is anomalous concerning the entire data set, it is termed as a collective anomaly. Data instances in a collective anomaly may not be anomalies by themselves, but their occurrence together as a collection is anomalous.”

3 Design and Development

This chapter of the dissertation will describe the design and development of the four artifacts and focus on the case study of the "Profit and Loss" forecast.

This chapter is organized in the following Sections:

- Section 3.1 – It is presented the case study and data that was used in this investigation.
- Section 3.2 –It is detailed the development language and the most important packages that were used.
- Section 3.3 – It is presented the first artifact, the instantiation of the baseline forecasting model.
- Section 3.4 – It is presented the first approach to the second artifact, the forecasting model. It was used Prophet and Theta Forecasting methods.
- Section 3.5 – It is presented the first approach to the second artifact, a meta-forecasting model with two forecasting methods.
- Section 3.6 – It is presented the third and final approach to the second artifact, the forecasting model, with a meta-learning forecast with six forecasting methods;
- Section 3.7 – It is presented the fourth and last artifact, the anomaly detection model.

3.1 Case Study

PRIMAVERA is a Portuguese technology company that has established itself in the national market of computer management solutions for being a pioneer in the development of Windows applications. It has offices in Portugal, Spain, Angola, Mozambique and Cape Verde, and an international network of 600 certified business partners and 40,000 customers across 20 countries.

In 2017, PRIMAVERA registered a business volume of 23.4 million euros, a value that corresponds to an increase of 14% in relation to the value reached in 2016.

For the present work, the data used were the total monthly "Profit and Loss" of about 6000 duly anonymized companies. A forecasting method developed in Primavera BSS that tests the fit of the history of a set of classical methods was also used as a comparison method to use. We received 2 text files in the CSV format, one with the input of the forecast algorithm (Figure 7) and the other with the output (Figure 8) of the forecasting algorithm.

```
"Partition","Data","FirstYear","FirstMonth"
"1000228497","0, 949.03, 295, 604.5, 800, 680",2018,9
"1000587118","20.32, 41.46, -29.27, 0, 0, 0, 0, 0, 0, 0, 0, 0",2018,2
"100065431","4394.6, 0, 0, 0, 541.8, 0, 0",2018,8
"1001304770","0, 34100, 63624.8, 37602.98, 138740.84, 127102.87, 150169.3, 66666.67, 100213.82, 66943.09, 104674.4, 120116.26, 0",2018,2
"1001433266","0, 1471, 4134.3, 2461, 0, 1035, 3216.07, 1041.48, 0, 0",2018,5
"1001773576","0, 0, 100, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 100, 0, 0, 0",2017,7
"1001975075","3488.94, 0, 0, 0, 0, 0, 0, 0, 0",2018,6
"1002141354","3300, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 7855.68, 0, 0",2017,12
"1002283174","1784.88, 2522.56, 1918.21, 3914.92, 2958.26, 2095, 2666.67, 2025, 2255, 1818.33, 2125, 4616.3, 769.43, 1752.13, 1194.39, 3346.83, 2045.08, 0,
0, 0, 0, 0, 0, 0, 0, 0",2017,1
"1003352859","1065, 1860, 2205, 1625, 970, 410, 0, 435, 410, 435, 435, 0",2018,3
"1003494641","70, 0, 0, 40, 1314.4, 767.5, 0, 773",2018,7
"1004000335","145, 32, 0, 445",2018,11
```

Figure 7 Text input sample of the baseline forecast method

Basically, the input file has a "Partition" key that anonymizes the source of the data, a comma-separated string with a sequence of values that represent the monthly time series, and finally a first-year and first-month value that allow us to link each value to a date. This time-series represents the monthly value of Profit and Loss KPI.

The output is a text file in the CSV format. To get the forecasting result of the Baseline method we filtered the dataset with state equals to Success and parsed the comma-separated string forecast.

```
Partition;State;Message;Debug;Stats;Model;Accuracy;Confidence;Sale;Total;TotalLabels;Forecast;ForecastLabels
1000228497;Skipped;Insufficient number of data periods.;;6;;;;;0, 949, 295, 604, 800, 680;;;
1000587118;Skipped;Insufficient number of non-zero data periods.;;13;;;;;0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0;;;
100065431;Skipped;Insufficient number of data periods.;;7;;;;;90, 0, 0, 0, 542, 0, 0;;;
1001304770;Success;Successful forecast.;;13;Mean;100;113937;0;0, 34100, 63625, 37603, 138741, 127103, 150169, 66667, 100214, 66943, 104674, 120116, 0;2018-
02, 2018-03, 2018-04, 2018-05, 2018-06, 2018-07, 2018-08, 2018-09, 2018-10, 2018-11, 2018-12, 2019-01, 2019-02;77689, 77689, 77689, 77689, 77689, 77689,
77689, 77689, 77689, 77689, 77689, 77689;2019-03, 2019-04, 2019-05, 2019-06, 2019-07, 2019-08, 2019-09, 2019-10, 2019-11, 2019-12, 2020-01, 2020-02
1001433266;Skipped;Insufficient number of data periods.;;10;;;;;0, 1471, 4134, 2461, 0, 1035, 3216, 1041, 0, 0;;;
1001773576;Skipped;Insufficient number of non-zero data periods.;;20;;;;;0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0;;;
1001975075;Skipped;Insufficient number of data periods.;;9;;;;;0, 0, 0, 0, 0, 0, 0, 0, 0;;;
1002141354;Skipped;Insufficient number of non-zero data periods.;;15;;;;;0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0;;;
```

Figure 8 Text output sample of the baseline forecast method

3.2 Development Language

Python and R are the most common programming languages in Data Science. We opted to use Python as the main programming language because it is the most popular language and not only in Data Science projects. But, specifically for Time Series Forecasting, Rob Hyndman Forecast package in R is the most complete, so we used Rpy2 to allow us to use this R package.

We used the Anaconda Distribution to manage the Python packages and stated with Spyder IDE but changed to PyCharm mainly because of the autocomplete and code organized features.

Main Python packages that were used: - Pandas; Numpy; Keras; Tensorflow; Scikit-learn; PyOD - Outliers Detection; TS Fresh - Extract Features from Time Series; Fbprophet - Facebook Prophet; Rpy2 - Run R code in Python; Dask – Advances parallelism for analytics R – Forecast

3.3 Baseline

As a baseline, we used a forecasting method that was developed in Primavera BSS.

This forecasting method developed in R that tests the fit of the history of each time series in a set of classical and choose the one with the best fit to perform the forecast.

This method ignored timed series whose length was lower than six months because it was considered the need for a set of minimum historical values to increase the accuracy of the forecast.

A file with the output of the best fit forecast methods is produced.

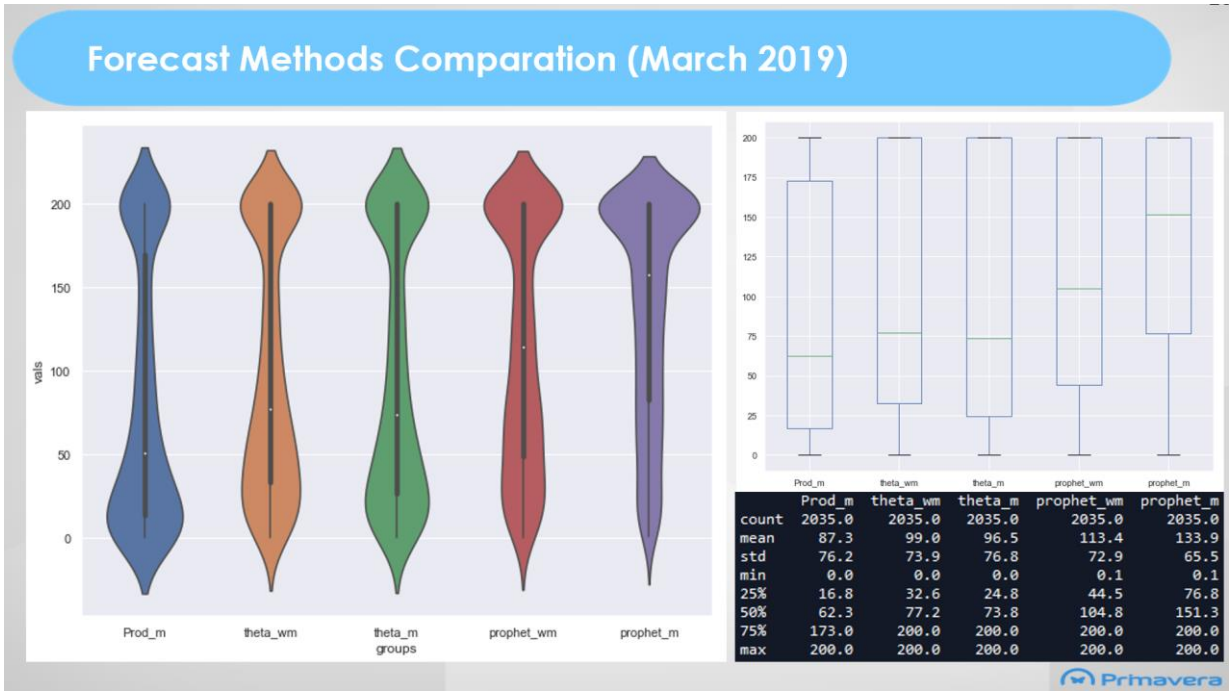
3.4 Prophet and Theta Forecasting Methods

In the first approach to the problem, two prediction methods were used, Prophet amd Theta, comparing the accuracy of each method and the baseline. In addition to the monthly Profit and Loss data, the same data was also tested, but using a weekly aggregation.

The Prophet Method is developed by Facebook Research and defined as "a model regression model with interpretable and intuitively adjustable parameters." It was presented in the paper "Forecasting at Scale" that motivated this investigation.

The Theta method is described as a version of Exponential Smoothing, having created academic interest for the performance achieved in the M3 competition

To visualize and present the distribution of the forecast accuracy metric (SMAPE), we used a mix of descriptive metrics, boxplot, and violin plot like it is presented in Figure 9. The lower value represents a greater accuracy, thus a method with better forecasting ability.



	Prod_m	theta_wm	theta_m	prophet_wm	prophet_m
count	2035.0	2035.0	2035.0	2035.0	2035.0
mean	87.3	99.0	96.5	113.4	133.9
std	76.2	73.9	76.8	72.9	65.5
min	0.0	0.0	0.0	0.1	0.1
25%	16.8	32.6	24.8	44.5	76.8
50%	62.3	77.2	73.8	104.8	151.3
75%	173.0	200.0	200.0	200.0	200.0
max	200.0	200.0	200.0	200.0	200.0

Figure 9 Detailed Forecast Comparation Methods of March 2019 data.

Both the violin plot and the boxplot show the distribution shape, the median, and the interquartile range. This graphic representation allows us to have much more information about the performance of the forecasting method compared to a simple mean.

The distribution shape allowed us to understand the distribution shape changed when we simple performed a filter. This filter removed all the time series where the observed value we were trying to predict was zero. We concluded that this time series represented companies that are no longer using the software and are negatively influencing the results (Figure 10)

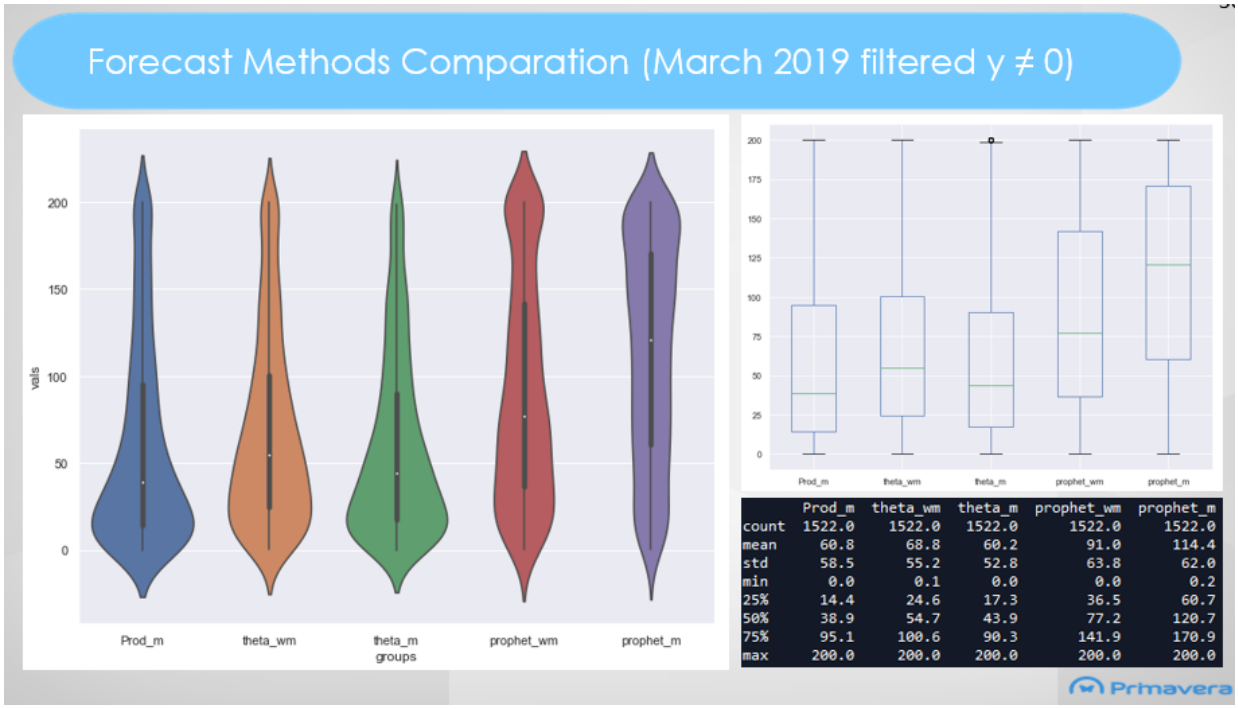


Figure 10 Detailed Forecast Comparation Methods of March 2019, filtered to remove all-time series where the observed value that we were trying to predict was zero.

By using this data visualization step, we were able to conclude that the removed time series were contributing to performance degradation of the forecasting methods.

In Figure 11, it is summarized the accuracy results (SMAPE) of the methods used for the March 2019 forecast. The mean error (SMAPE) was calculated.

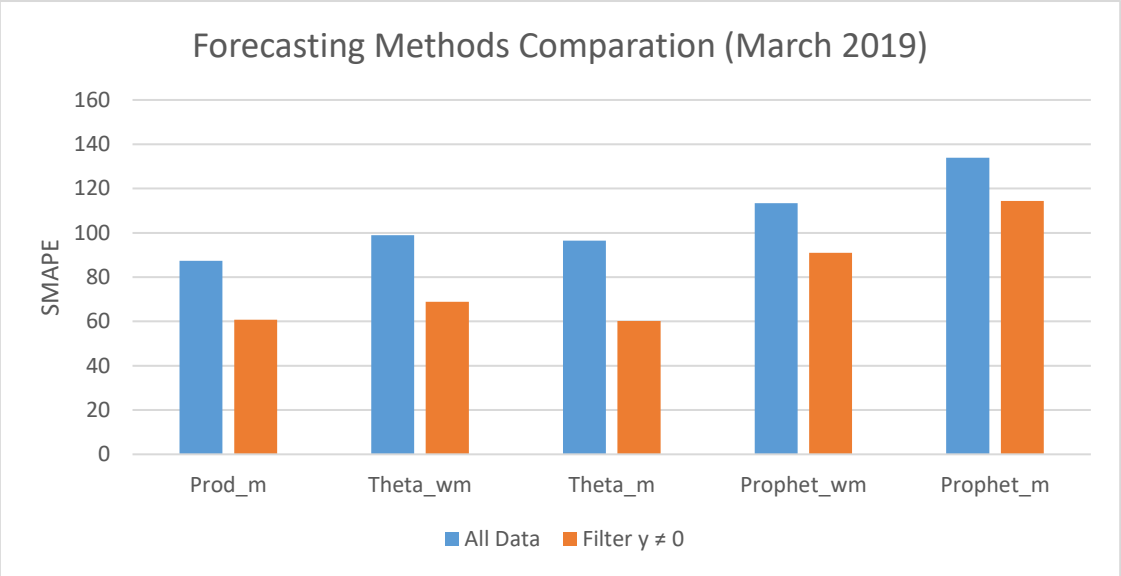


Figure 11 Forecast accuracy comparison (SMAPE) between Prophet and Theta in a month and week to month data

The Theta and Prophet method was tested with monthly data (m) as well as weekly data resample to month (wm), so we could compare the results. To be able to compare the results, only the predictions calculated by the five methods were used. There are two series, the blue one that has all the results (2035 forecasts), and the orange one was filtering to the companies that did not have sales in the month under study (result in 1522 forecasts).

In this first approach, it is concluded that under the test conditions, the Theta method is the closest to the baseline (production method), slightly exceeding the orange series monthly forecasts. On the other hand, forecasts with weekly data, although with better results in the prophet method, the same is not true in the Theta method. Finally, it is concluded that the results improve substantially when we do not consider companies without results in March (orange).

3.5 Meta-Learning with Prophet and Theta

In a second approach, we opted to test the possibility of training a classifier to identify the best forecasting method, taking as its starting point a set of features taken from the time series (Meta-Learning). To use this method, a field calculated with the best method was added, which represents the minimum error calculated by prediction in the Theta and Prophet methods.

For input (X) in the classifier, about 700 time-series features were obtained. It was used tsfresh package which "automatically calculates a large number of time series characteristics, the so called features" and "contains methods to evaluate the explaining power and importance of such characteristics for regression or classification tasks". (Maximilian) From these features we selected those that had a higher correlation with the SMAPE error of each of the methods under study (about 40). For the result to be obtained from the classifier (Y), a new categorical field representing the best method was added. After dividing into Training and Testing, three classifiers were tested, Random Forest, Logistic Regression, and SVN, with Random Forest achieving the best result with an average accuracy of 60%.

In Figure 12 we present a scheme that summarizes the train and classification of the Random Forest Classifier used to predict the best forecasting method for each time series.

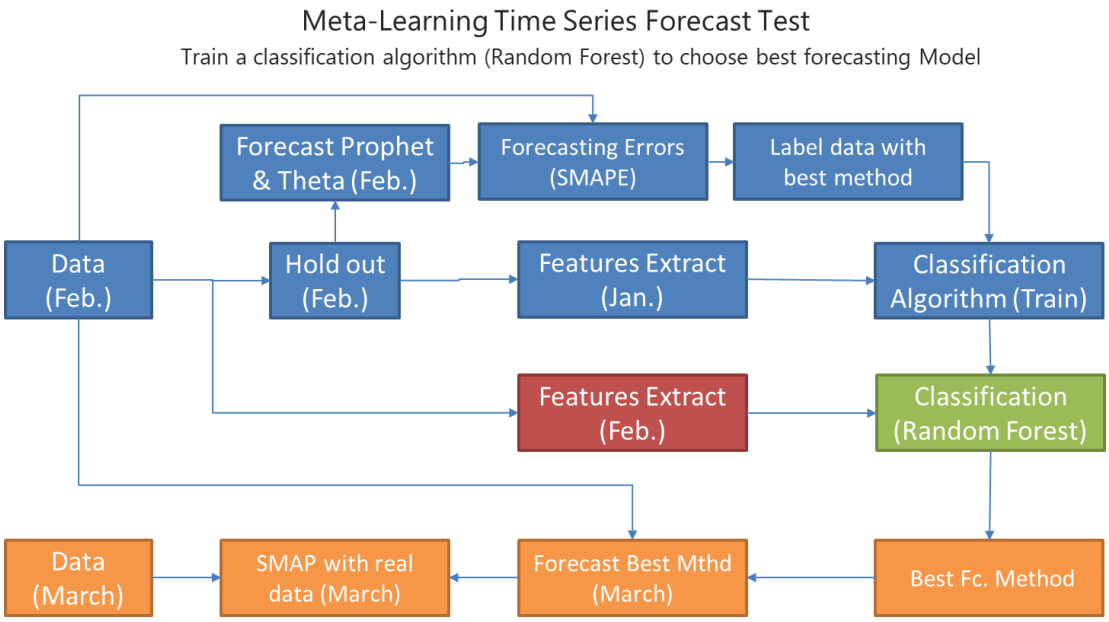
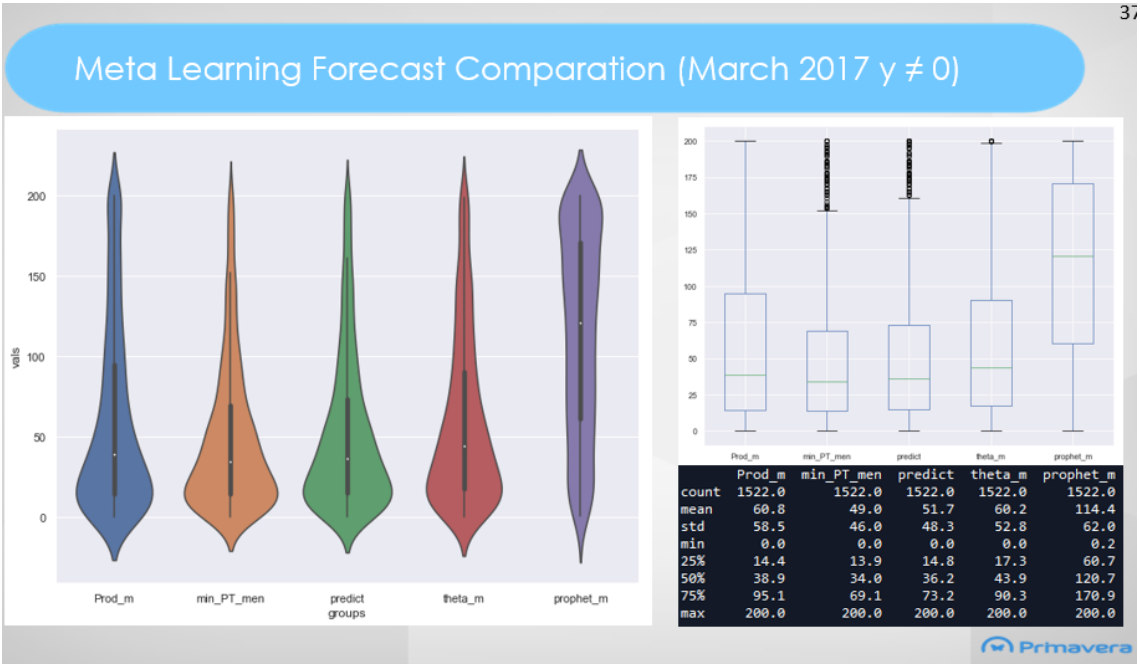


Figure 12 A meta-learning prototype test scheme

Stating in the "Data (Feb.)" represents "Profit and Loss" until February 2019 (included). Then this month was removed (Hold out), and it was calculated the Forecast for February using both Prophet and Theta Method and calculated the Forecasting error by comparing the forecast value with the real value. Then we label the data with the best forecasting method. The labeled dataset and the features extracted were used to train a Random Forest classifier. Then we tested this classifier with the features extracted from the original dataset (Data Feb.) that allow us to determine what was the theoretical best forecasting method for each time series. At last, using the Profit and Loss of March, we were able to calculate the forecasting error (SMAPE) of March.

The results of the Meta-Learning classification can be viewed in Figure 13 and Figure 14 with the label "Predict".



	Prod_m	min_PT_men	predict	theta_m	prophet_m
count	1522.0	1522.0	1522.0	1522.0	1522.0
mean	60.8	49.0	51.7	60.2	114.4
std	58.5	46.0	48.3	52.8	62.0
min	0.0	0.0	0.0	0.0	0.2
25%	14.4	13.9	14.8	17.3	60.7
50%	38.9	34.0	36.2	43.9	120.7
75%	95.1	69.1	73.2	90.3	170.9
max	200.0	200.0	200.0	200.0	200.0

Figure 13- Detailed Forecast Comparation Methods of March 2019 between baseline (Production) Theta and Prophet forecast methods and meta-learning prototype test scheme (Predict) filtered to remove all-time series where the observed value that we were trying to predict was zero

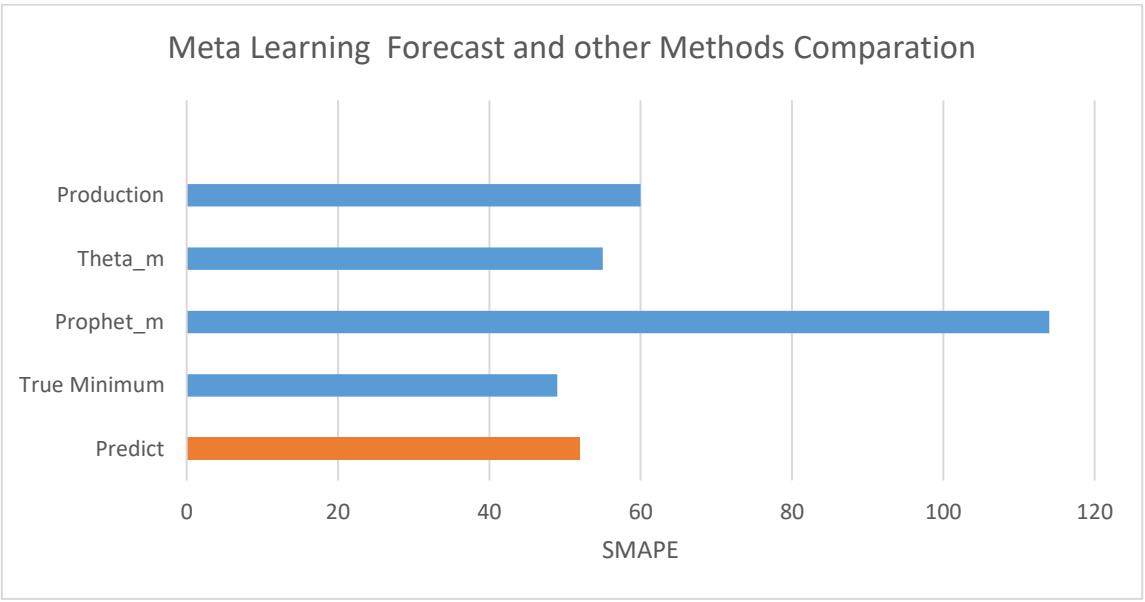


Figure 14 Forecast accuracy comparison (SMAPE) between Baseline (Production) Theta and Prophet forecast methods and meta-learning prototype test scheme (Predict). It this prototype it was removed all-time series with trailing zeros resulting in 1522 time series.

These results allow us to conclude that the approach of trying to find a forecasting method that would perform well in any time series could not be the most suitable. Just as there is no magic medicine that cures all diseases, there is neither a magic forecasting method. So, we must try to choose the best forecasting method for each time series.

Our classifier is like a doctor who prescribes a specific drug to try to cure a certain disease.

In the third and last approach, we worked with six forecasting methods to try to obtain better results.

3.6 Meta-Learning with 6 forecasting Methods

For the third and last approach, we work with 6 forecasting methods to apply the meta-learning method previously describe. We kept the Thetaf method but dropped the Prophet as the results weren't as good as expected. We add two basic forecasting methods: rwf and meanf (Table 6), and to Theta method used before we add ARIMA and two versions of Exponential Smoothing (Table 7).

Method	Description
RWF	Forecast of all future values are equal to the value of the last observation
Meanf	Forecasts of all future values are equal to the average of the historical data

Table 6 Simple forecasting method used in meta learning

Method	Description
Auto-ARIMA	Tries to automatic obtain the best argument for the ARIMA model (p,d,q)
ETS	Error, Trend Seasonal the Exponential Smoothing method
ETS_Damped	Adds a "damped trend" to ETS
Thetaf	Exponential smoothing with drift

Table 7 Other forecasting method used in meta learning

Working with six forecasting methods, the time it took for the prediction algorithm to run began to be a concern. In order to drop this time, we add a Python parallel computing library that allows us to drop the forecasting time to 30% (from 50 minutes to 15 minutes).

The results of the forecasting method are shown in Figure 15. Rwf which simply repeats the last observation gets the best results but suffers from high degradation as the forecast period increases for 3 to 6 months. Auto Arima gets the second-best result.

To train the classifier we have used a combinatorics analysis. A combination of 6 methods 2 and 3 at a time was calculated. For the combination two at a time, meanf and rwf got a better result, and for three at a time meanf, rwf and thetaf got a better result.

We will discuss the results of this final approach with more detail in the next chapter.

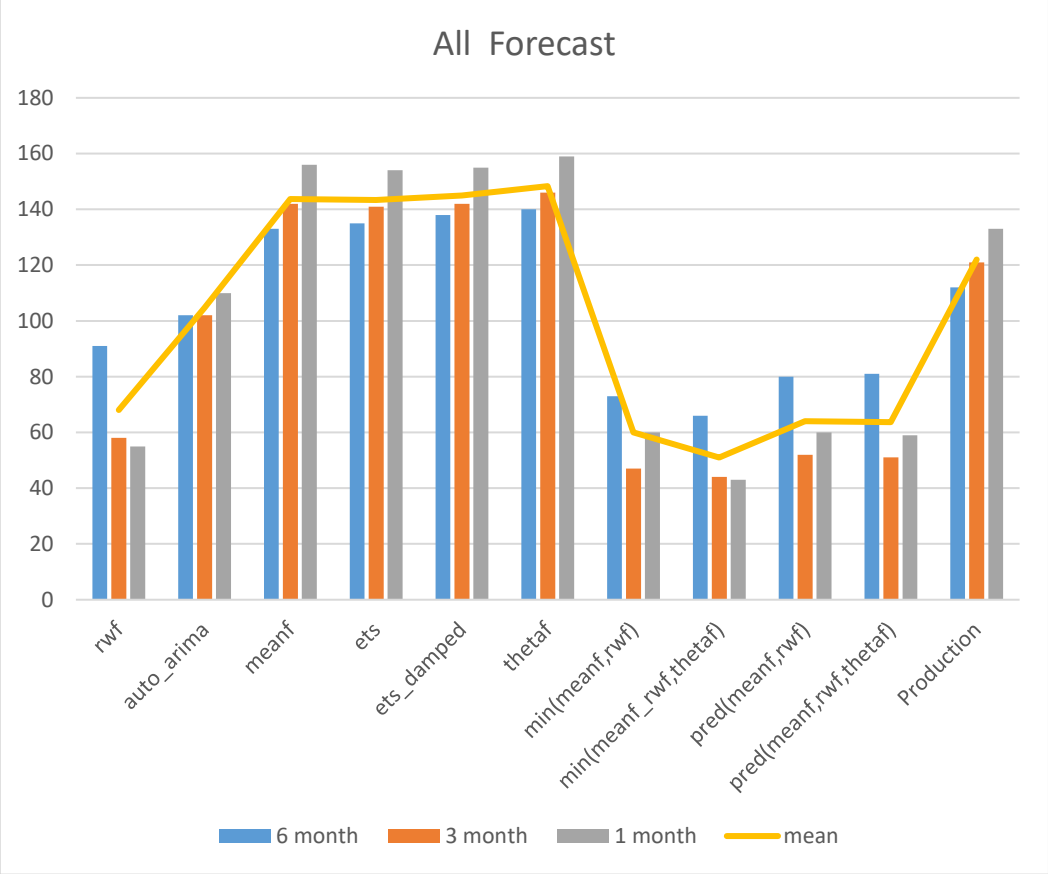


Figure 15 Results of Meta-Learning with 6 forecasting Methods

3.7 Anomaly Detection

As defined before in Section 2.7 Prediction Interval is an interval within which we expect the future observation (that was forecasted) to lie. The prediction interval is calculated using a specified confidence level, usually 95% and/or 80% (Hyndman, et al., 2019) Section 3.5. It was assumed that errors are normally distributed.

Some forecasting models like Auto-Arima allow us to get the prediction interval, for a specified probability in the forecast period. This allow us to check, as new data becomes available, if it is within the expected interval (Prediction Interval - the expected behavior), or outside the interval which can be label as a possible "anomaly".

In Figure 16 we can see an example of the prediction interval. The space in February 2019 divides the historical data (left side) and the forecast data (right side). The forecast values were purposely omitted, but it was plotted the "real values after the forecast" where we can apply the anomaly detection. Two point in March and June 2019 were detected as local anomaly points and are marked as a red dot.

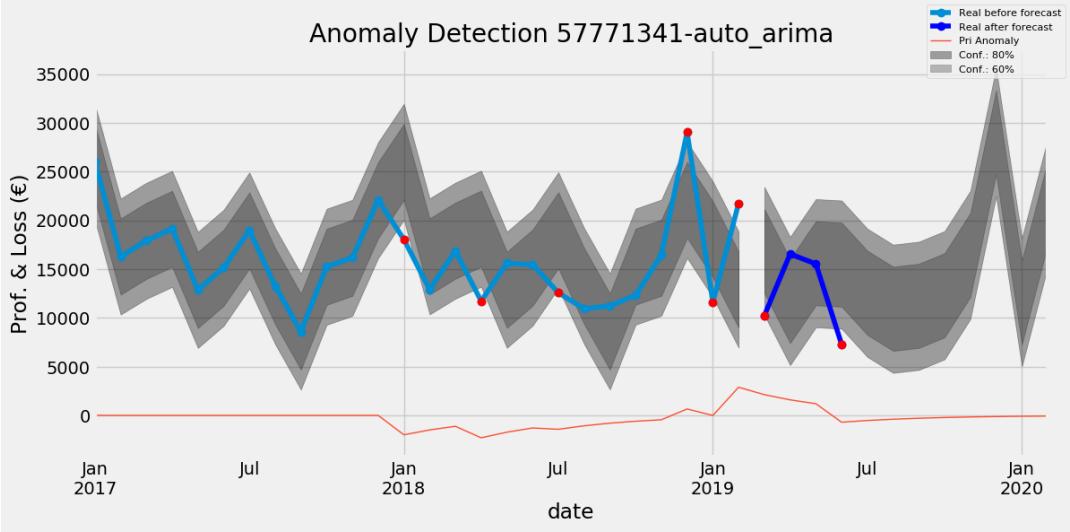


Figure 16 An example of the historical values of a time series and the forecast with auto-arima. The prediction interval is calculated in historical and forecast data with two probabilities (80% and 60%)

In Figure 16 we show the prediction interval in the historic data too. This interval is useful to calculate the anomaly history and allow us to easily understand where our model would detect anomalies in the historical data.

The prediction interval in the historic data is not directly obtained by the forecast method. To calculate it, for example with 80% confidence level we use the fitted value in

$$\hat{y}_t \pm 1{,}28\hat{\sigma}_h$$

where $\hat{\sigma}_h$ is the estimated standard error of the residuals that can be obtained directly in the forecasting method.

As we can see in Figure 16, all points outside the prediction interval are marked as possible "local" anomalies with a red dot.

The prediction interval and the marked anomalies add useful information that helps the analysis of the past and predicted behavior of the time series. But if we want a solution to detected anomalies at scale, it is expected to alert for the biggest anomalies as an alert for every anomaly would generate a high number of anomalies. This led us to the need to measure the anomalies.

To measure anomalies, we calculate de distance between the point outside the prediction interval and the closest boundary of the prediction interval. We call it a local anomaly. Local anomalies above the prediction interval have a positive signal, and below de prediction interval have a negative signal.

We then used an accumulator in order to capture the history of anomalies. The main reason is that a sequence of small anomalies can be as important as one big anomaly. In other hand, a typical scenario when someone makes a mistake (like duplicate an invoice) and later corrects that mistake (opposite transaction), there will be two anomalies with the same absolute value, but one positive and one negative that will cancel each other in the accumulator (the mistake is fixed).

As we wanted that recent anomalies were valued compared to older anomalies, we create a "forget factor". This factor will approximate its weight value to zero as time goes by. By using this factor, we know that with anomalies with the same weight but that happens in two different time points, the most recent will have a greater weight.

Our anomaly detection method has a different approach from the methods explain in the related work. Our meta-learning forecast model can produce a large number of forecasts, and an expected interval within which we expect the future observation to lie. We expect that business decision will be made based on the forecast, and our anomaly detection method will alert in the ingestion of new data if the behavior is far from the expected "previously forecasted" behavior. By doing it we are transforming a possible "forecast disaster" in a "intelligent alert of an unexpected behavior".

4 Demonstration and Evaluation

This chapter of the dissertation will describe the Demonstration and Evaluation of the four artifacts.

This chapter is organized in the following Sections:

- Section 4.1 – Presents the results of the baseline forecasting method.
- Section 4.2 – Presents the results of the proposed forecasting method and compares the results with the baseline forecasting method.
- Section 4.3 – Presents the results of the anomaly detection model.
- Section 4.4 – Presents an additional evaluation of the anomaly detection in a synthetic dataset.

4.1 Features and Visualization

For having a better knowledge of our dataset, we used some visualizations. In Figure 17 we can see an example of a time series randomly selected. In Figure 18 we can see an example of a decomposition of a time series in Trend, Seasonal and Residual (error) using an additive model of frequency twelve (that represents the twelve months of the year).

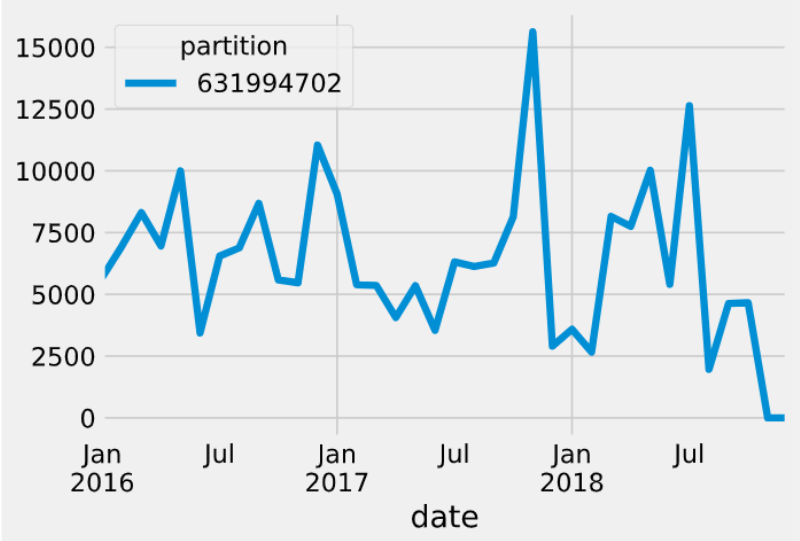


Figure 17 Plot of an example input time series.

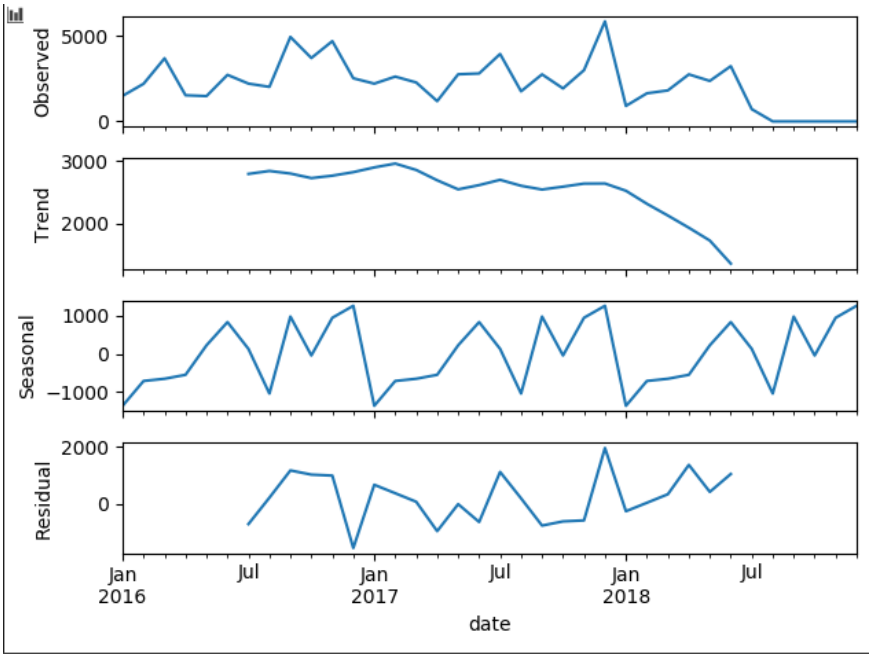


Figure 18 - Time Series Decomposition Example: sm.tsa.seasonal_decompose df_plot , model='additive', freq =12

We study the number of observations in the time series. In Figure 19 we can see that our time series have less than 24 month long (with an outlier in the 35 month) but there is a very high incidence in series under 15 months

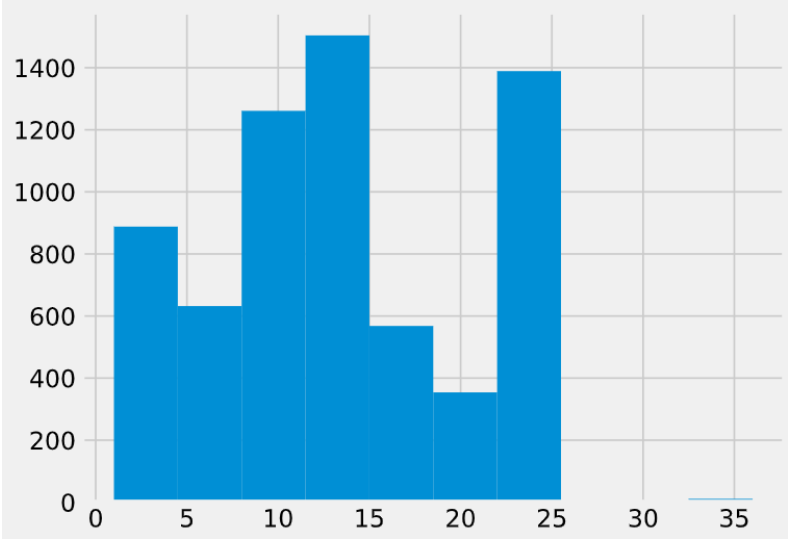


Figure 19 Distribution of the count of the total number of observations in each time series.

We study the feature distribution to see if we could find any pattern in the data. In Figure 20 we can see the density of the time series feature "strength of seasonality" where we can conclude that our data, in general, has a low strength in this feature.

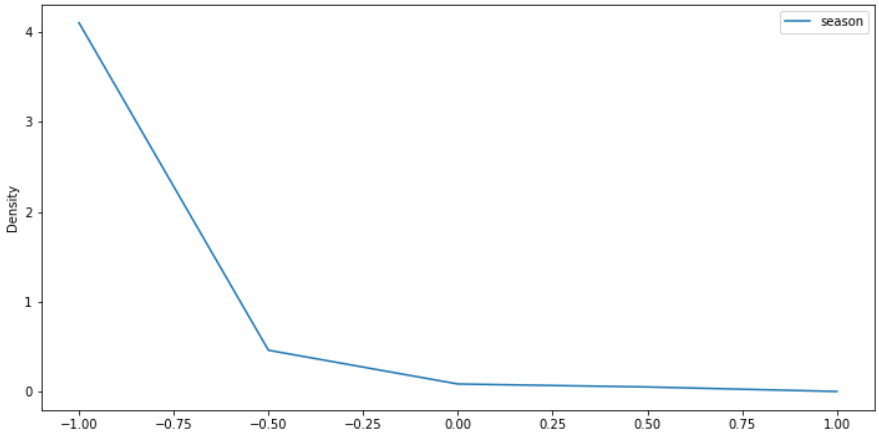


Figure 20 Distribution of Strength of Seasonality, low seasonality (-1) to high seasonality1

To better understand the difference between a low and a high seasonality strength, we plot two examples of two time-series, one that represent a low strength seasonality (Figure 21) and other that represent a high strength seasonality (Figure 22). We choose a plot by month with each year in a separated series to better understand the seasonality effect.

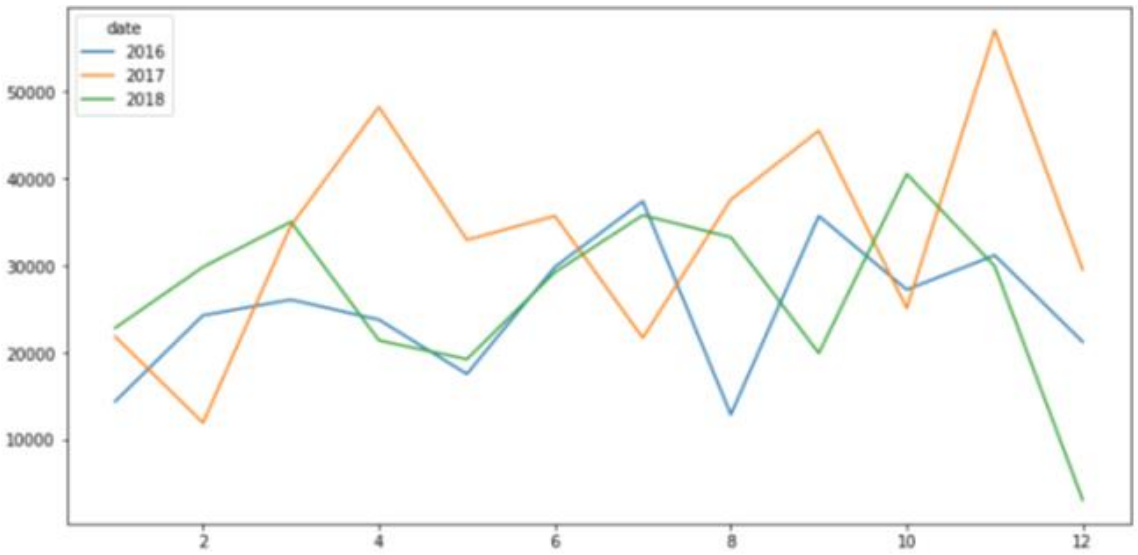


Figure 21 Feature Strength of Seasonality, an example of a low Seasonality

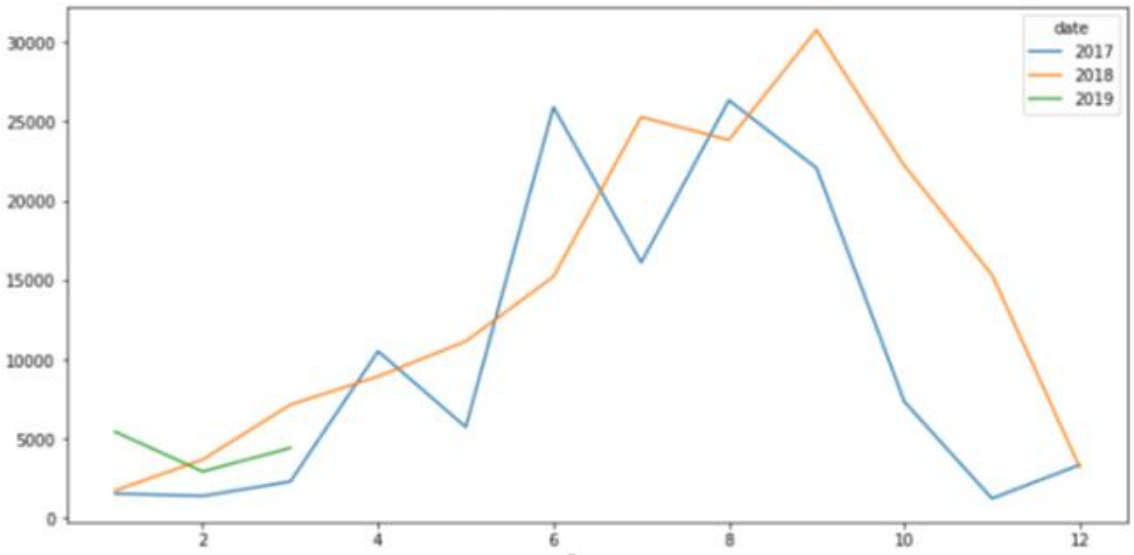


Figure 22 Feature Strength of Seasonality, an example of a high Seasonality

It is clear that in Figure 22 the behavior of the time series is clearly affected by the time of the year, where there is an increasing sales performance between the start of the year and September, followed by a decrease until the end of the year.

We plot a feature behavior matrix where we could compare de estimated density of each time series feature and a comparative distribution of two features. And smaller example of this analysis is presented in Figure 23.

We were looking to find linear or nonlinear relations like what is found between ACF1 (auto-correlation function) and diff_acf1. In this case, this is an expected relation because they are two related features.

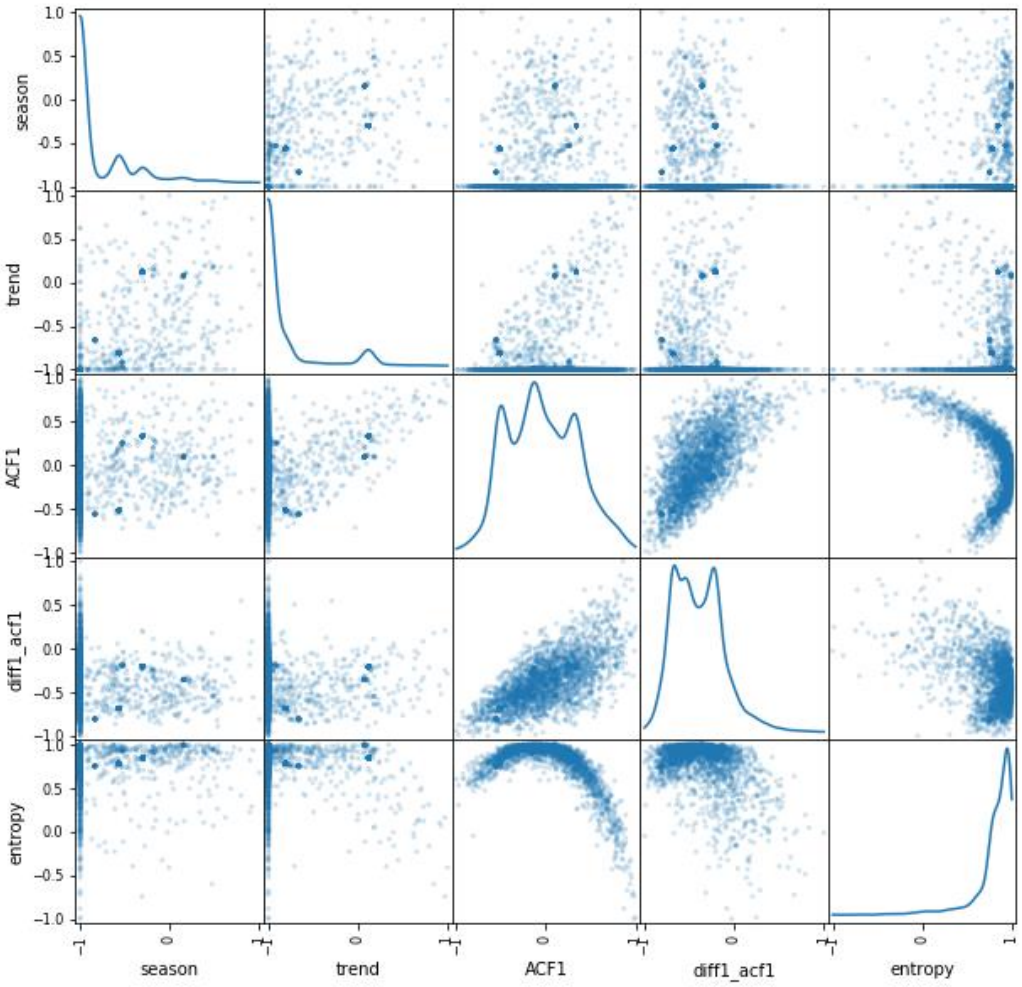


Figure 23 – Example of Time Series Features study with feature distribution.

Looking at the main diagonal of the matrix, the first graph represents the distribution of the strength of the seasonality and the second the distribution of the strength of the trend. In both distributions the greatest weight is near left margin which represents a low trend/seasonality.

4.2 Baseline Forecast

The baseline forecast method was running once a month and forecasting twelve months of the "Profit and Lost" KPI using an R algorithm triggered in an Azure Data Factory.

As was explained in the previous chapter, a set of 6 forecasting methods are used and it is selected the forecast method that successful runs and have the closest fit to the history of the time series. An outlier detection is performed, and observations marked as outliers are removed and replaced by the average of the time-series observations. The applied algorithm mark as outlier observations that lie outside the interval formed by the value 1.5 multiplied by IQR. IQR is the Inter Quartile Range which is the difference between the 3rd and the 1st quartile (or 75th and 25th percentile).

There was defined a pre-condition in the input of the forecasting algorithm. A time series must have a minimum of 12 data points (to get an advantage of an expected twelve-month business seasonality), and at least 6 non-zero data points (considered a irregular customer) in order to be forecasted.

4.3 Proposed Forecasting Method

We propose a forecasting method that uses a meta-learning classifier to identify the best forecasting method. As explained in the previous chapter, the classifier uses a set of features extracted from the time series and a dataset that was labeled with the best method to train this classifier.

As shown in Figure 24, we tested three different classifiers methods: SVN, Logistic Regression and Random Forest Classifier. The Random Forest Classifier got the best accuracy result with 60%.

In Figure 24 it can be seen the results of the three tested classifier methods.

Model	Accurancy
SVN	51%
Logistic Regression	53%
Random Forest Classifier	60%

Figure 24 Results of the accuracy of three different classifiers methods

In our final propose, we opted not to use any outlier removal technique. We know that outliers will influence the forecasting, but they are part of the data and we think that it must be the decision marker's choice to manually opt to remove them.

We didn't use any pre-condition and tried to forecast every time series.

In Figure 25 and Table 8 it can be seen the results of the forecasting error (SMAPE) of one, three and six-month forecasting in the six forecasting methods (rwf, auto_arima, meanf, ets, ets_damped, thetaf). A yellow line was used to show the mean of the three forecasting periods. Being an error measure, a lower value is better.

Comparing the results of the six forecasting methods, we see that rwf and auto arima got the best results, and meanf, ets, ets_damped and thetaf seem to have very similar results.

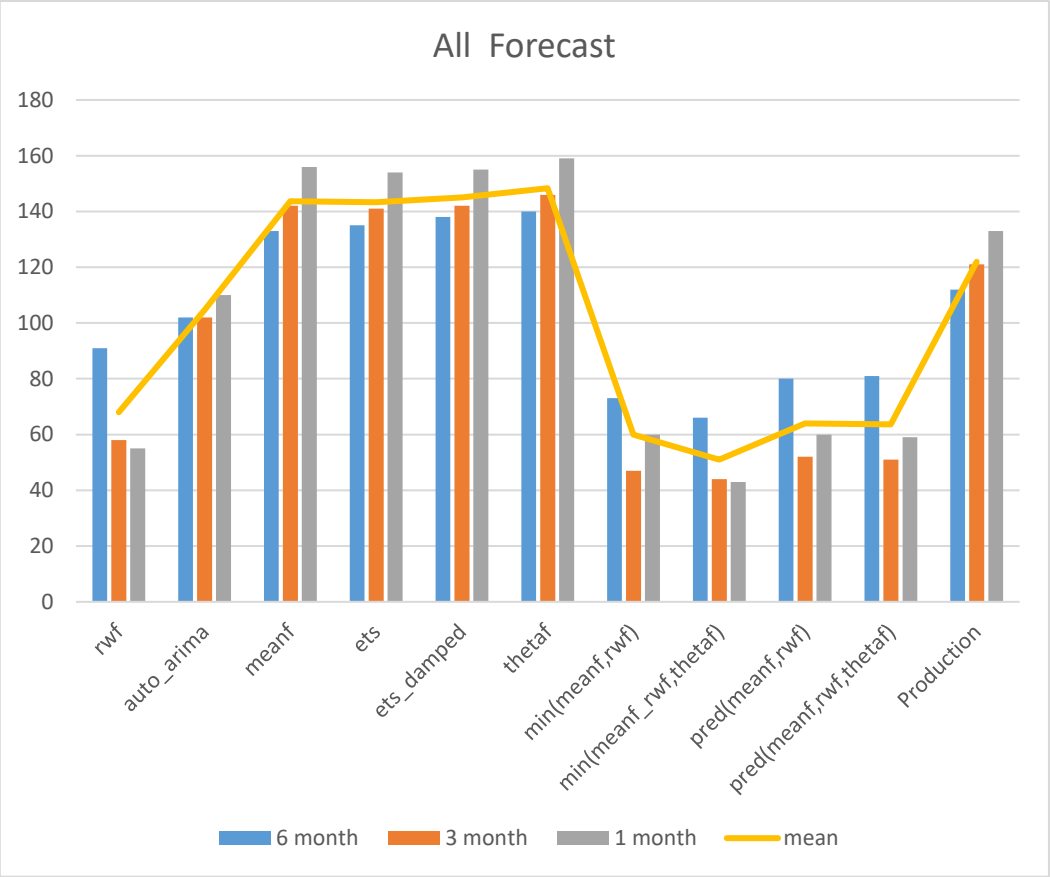


Figure 25 Final results of the forecasting error (SMAPE) in all used forecasting methods in one, three, six forecast periods and the mean of them. The theoretical minimum of the best combination of two and three forecasting methods, and classifier prediction intervals. The baseline (production) model is present but should be directly compared because it produces a significantly lower number of forecasts.

It is important to note that simply using the average error to compare 2 methods can be misleading because the methods may have completely different accuracy depending on the type of time series. To guarantee that we have the best results, we use combinatorial analysis where we found the best combination of two at a time and three of a time of forecasting methods which are "meanf, rwf" and "meanf,rwf, thetaf" respectively. The minimum shows the best theoretical result only possible if the classifier hit 100% and was obtained in the train data. The pred (prediction) is the result of the classifier with new data.

We plotted the baseline (production), but the intention is not to directly compare the performance of any forecasting method with the baseline. This is because, as it can see in Table 8, the cardinality (forecast count) of the baseline is much lower than other present methods. Next, we will remove from the results all the time series that aren't present in the baseline dataset. This will allow us to directly compare all methods.

	Forecast SMAPE				Forecast Count		
Model	**6 month**	**3 month**	**1 month**	**mean**	**6 month**	**3 month**	**1 month**
rwf	91	58	55	68	30427	17001	6095
auto_arima	102	102	110	105	30427	17001	6095
meanf	133	142	156	144	30427	17001	6095
ets	135	141	154	143	30427	17001	6095
ets_damped	138	142	155	145	30427	17001	6095
thetaf	140	146	159	148	30103	16827	6057
min(meanf,rwf)	73	47	60	60	30427	17001	6095
min(meanf_rwf,thetaf)	66	44	43	51	30427	17001	6095
pred(meanf,rwf)	80	52	60	64	26540	15031	5424
pred(meanf,rwf,thetaf	81	51	59	64	26510	15031	5424
Production	112	121	133	122	17973	8964	2964

Table 8 Final results of the forecasting error (SMAPE) in all used forecasting methods in one, three and six-month forecast. The theoretical minimum of the best combination of 2 and 3 forecasting methods, and classifier prediction intervals in bold (final result). The baseline (production) model is present but should be directly compared because it produces a significantly lower number of forecasts.

Finally, we removed from the results the time series that weren't present in the baseline dataset in order to directly compare the accuracy of each forecast method and the baseline method as it can be seen in Figure 26 and Table 9.

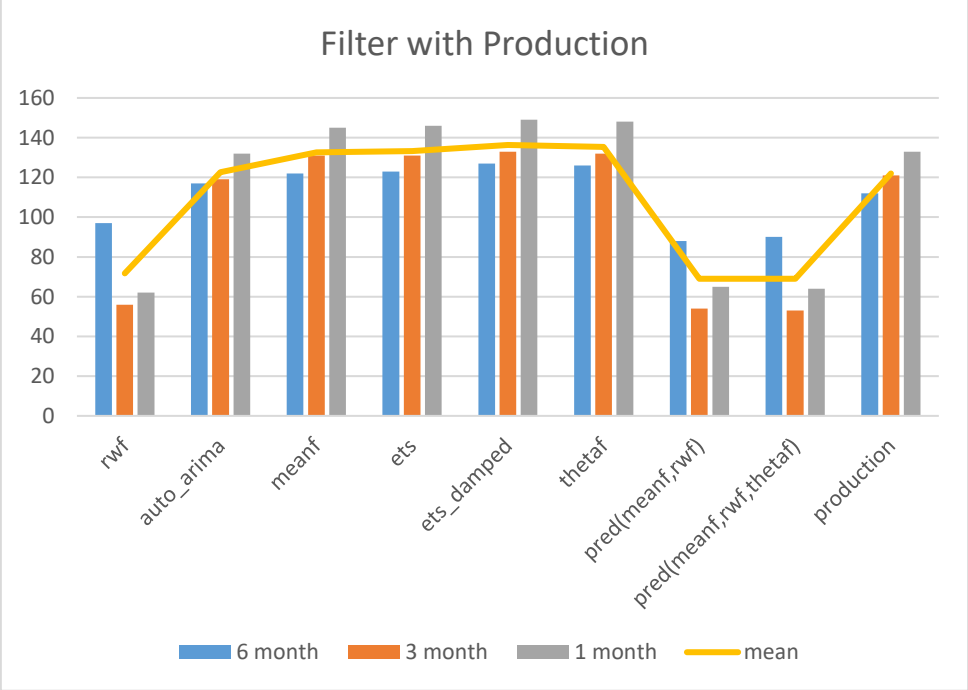


Figure 26 Plot of Final results of the forecasting error (SMAPE) in all used forecasting method filtered with the baseline (production) forecast output to allow a direct comparison of performance in one, two and six months forecast. It is present the theoretical minimum of the best combination of 2 and 3 forecasting methods, and classifier prediction intervals.

It stands out that individually, every forecasting method has the worst performance that the baseline, except rwf that simply repeats the last value. But our classifier using the

best combination of two and three at a time forecasting methods got a better result. The baseline SMAPE error dropped from the mean 122 to 69. And in three month period, dropped from 121 to 54.

	Forecast SMAPE				Forecast Count		
Model	**6 month**	**3 month**	**1 month**	**mean**	**6 month**	**3 month**	**1 month**
rwf	97	56	62	72	17973	8964	2964
auto_arima	117	119	132	123	17973	8964	2964
meanf	122	131	145	133	17973	8964	2964
ets	123	131	146	133	17973	8964	2964
ets_damped	127	133	149	136	17973	8964	2964
thetaf	126	132	148	135	17961	8958	2960
pred(meanf,rwf)	88	54	65	69	16972	8464	2798
pred(meanf,rwf,thetaf	90	53	64	69	16972	8464	2798
production	112	121	133	122	17973	8964	2964

Table 9 Final results of the forecasting error (SMAPE) in all used forecasting methods filtered with the baseline (production) forecast output to allow a direct comparison of performance in one, three and six months. It presents the classifier prediction intervals in bold (the result).

It can, therefore, be concluded that we can get much more forecasts with better accuracy than baseline (Table 8) and, on a direct comparison with identical datasets, we can also get better results than the baseline (Table 9).

We also note that the performance of predictive models, on most tested models, is better with six months forecast than three or one month forecast.

Next will see how we use the prediction interval to detect anomalies.

4.4 Anomaly Detection

By using the forecasting methods, we estimated the prediction interval not only to the forecast period but also to the historical period.

The red dots are anomalies when the data point (history or forecast) is outside the prediction interval. Like explained in the previous chapter, the local anomaly is measured by the distance between the point outside the prediction interval, and the closest boundary of the prediction interval. For each time series, we created a "Pri Anomaly" metric that is an accumulator of local anomalies with a forgetting factor.

In Figure 27 we can see an example of these historical and forecast periods with two prediction intervals of 60% and 80% confidence level, the local anomalies and the value of the Pri Anomaly metric.

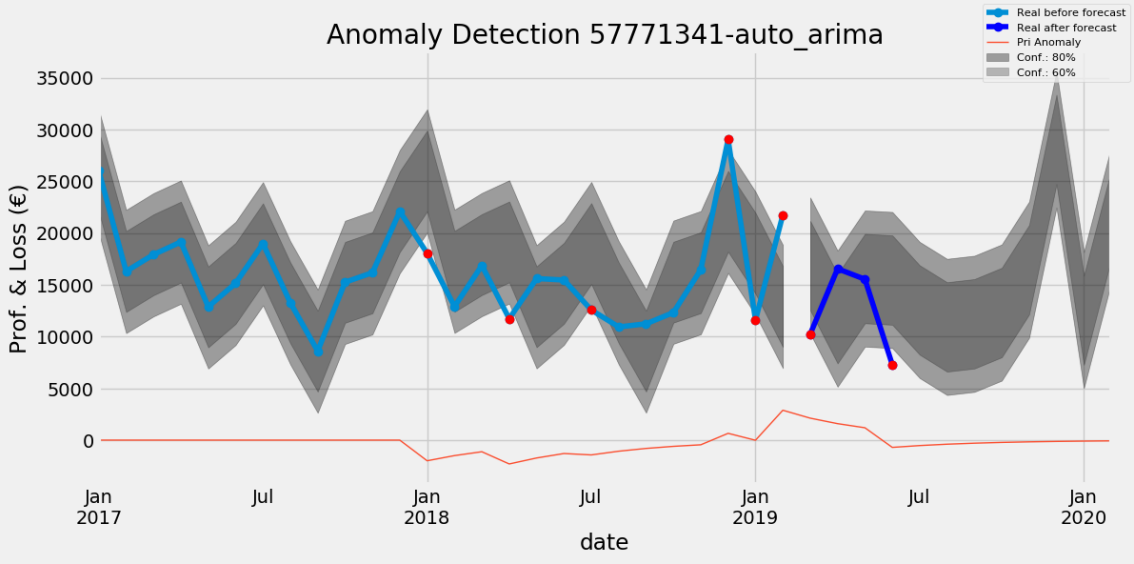


Figure 27 An example of the historical values of a time series and the forecast with auto-arima. The prediction interval is calculated in historical and forecast data with two probabilities (80% and 60%)

There are some benefits to using this technique to detect and measure anomalies. The anomaly is measured in the same unit that the data, and easily we can order all-time series by the Pri Anomaly metric that will highlight the time series with the highest accumulated anomaly. By ranking anomalies, we do not trigger a high number of alerts and the user can simply go how deep in the anomaly search as he wants. It's like a google search where it can be found millions of results, but we usually see the first 10 or 20 results. The signal of the "Pri Anomaly" Metric shows us if it is an accumulated positive anomaly (above the prediction interval) or an accumulated negative anomaly (below the prediction interval).

Another benefit of the "Pri Anomaly" metric is that it can be used not only to alert but also to trigger the forecasting method and the behavior of time series has changed, and we should consider the use of new data. This is important because we can skip the need for short scheduled forecasting if new data if keeping the except behavior.

To evaluate the precision of this anomaly detection and measurement, a team from Primavera generated a synthetic dataset with 18.000 time series with different types of anomalies. The results of this analysis are presented next.

4.5 Anomaly Detection in Synthetic Data

For evaluating the "Pri Anomaly" metric, a team from Primavera generated a synthetic dataset with 18.000 time-series. Many different types of anomalies were provoked and our challenge was to find the top 10 accumulated anomalies.

In this dataset, we did not perform any forecast. We used Auto Arima to get the prediction interval and detected and measure anomalies similarly we used in the historical data in the anomaly detection with the forecasting dataset.

We archived good results in our challenge, and the most important anomalies were found in seasonal and non-seasonal data.

We show four examples of time series anomalies.

In Figure 28 we have a non-seasonal time series, with a single anomaly in October 2016. We can see the “forget factor” effect next to the anomaly.

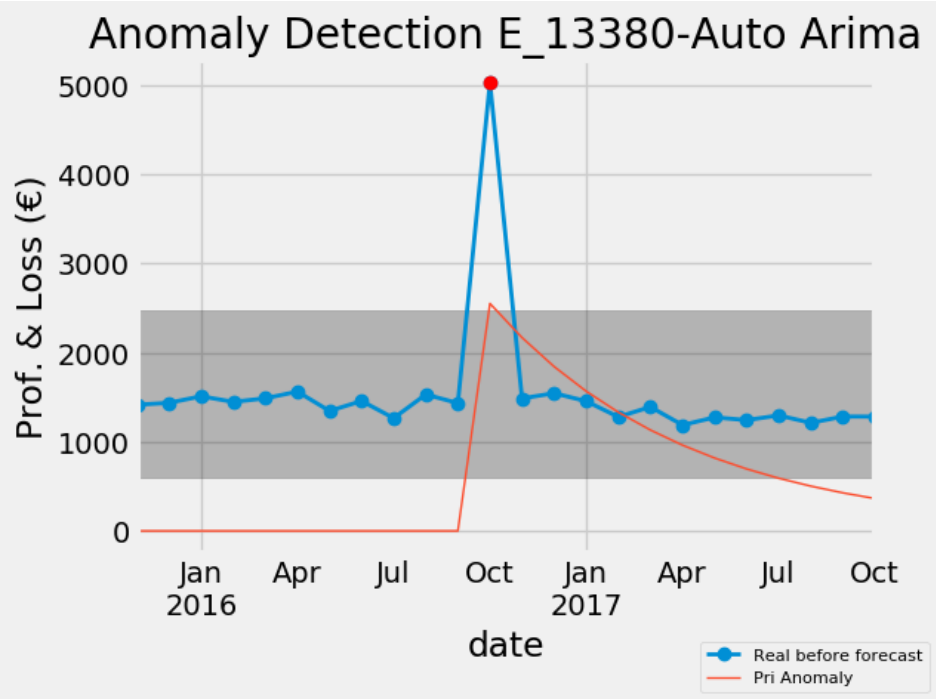


Figure 28 First example of the anomaly detection and measurement in the produced synthetic data

In Figure 29 we have another two years' time series, where we detect two major changes of behavior in July 2016 and October 2017. We can see the "forget factor" effect between August 2016 and October.

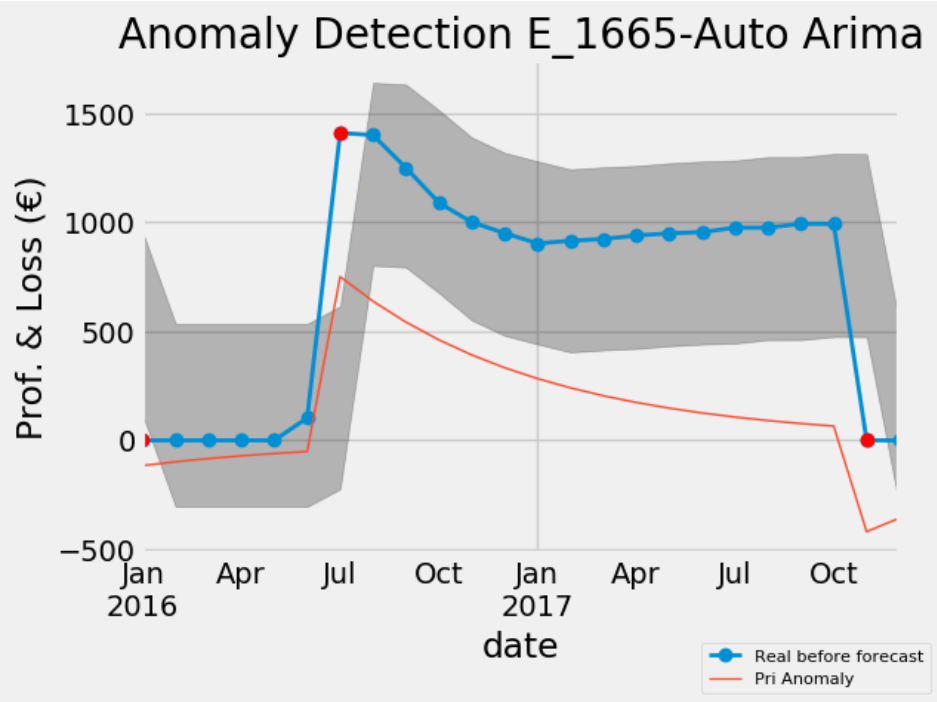


Figure 29 Second example of the anomaly detection and measurement in the produced synthetic data

In Figure 30 we use a seasonal time series with a change of behavior in February 2017. The anomaly in the first data point was discus but it was found useful as it will allow us to be alerted by the initial sale of the customer, as it is a change of the behavior of a non-buying customer to a buying customer.

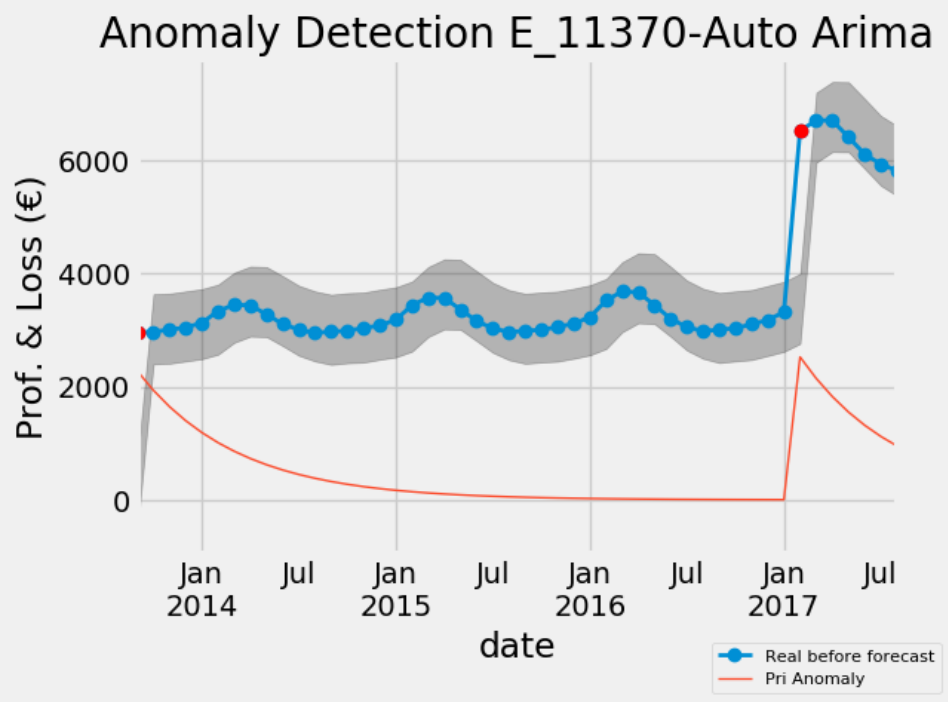


Figure 30 Third example of the anomaly detection and measurement in the produced synthetic data

Finally, in Figure 31 we see a noisy time series, where some hight values and zero data points are marked as anomalies. The behavior of the “Pri Anomaly” metric reflects these anomalies and will be helpful when reading this kind of charts

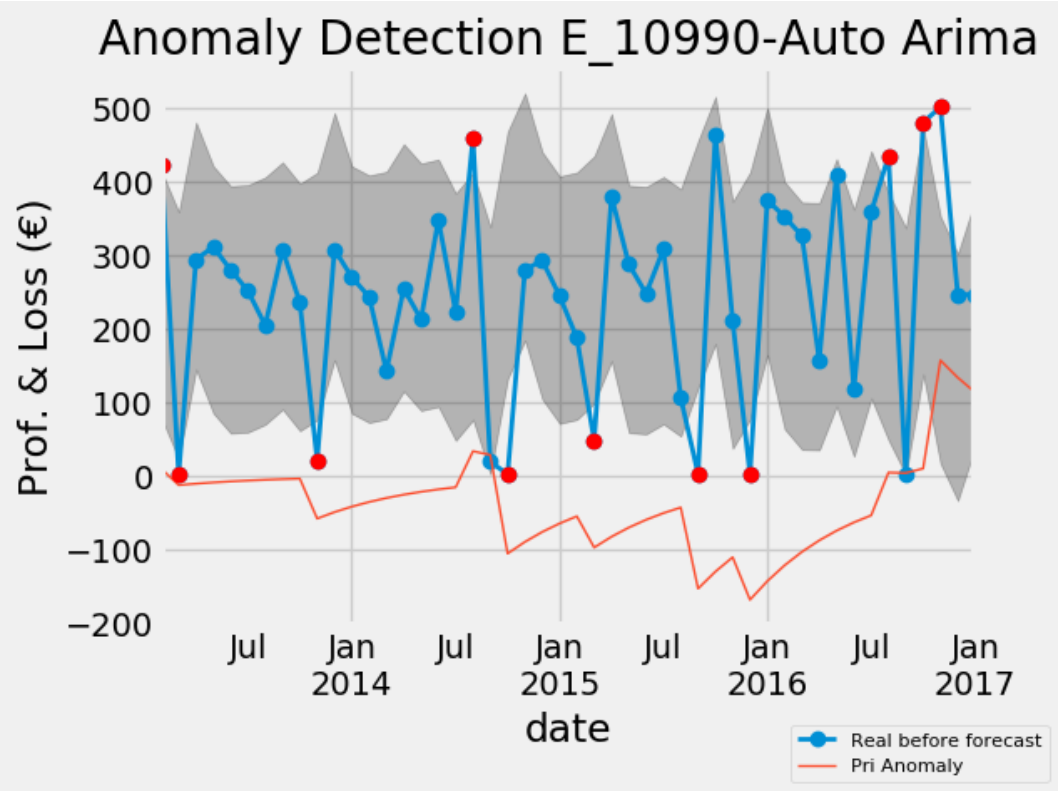


Figure 31 Forth example of the anomaly detection and measurement in the produced synthetic data

5 Conclusions and recommendations

4In this chapter we will describe the main conclusions of our study, the limitations we had to deal with, and propose future research.

This chapter is organized in the following Sections:

- Section 5.1 – It is presented the conclusion of this investigation
- Section 5.2 – It is presented the study limitations
- Section 5.3 – It is proposed a future research

5.1. Conclusions

This research allows us to understand the challenges in the implementation of machine learning at scale in small companies..

It is important that small companies understand the value of the data they already have, and opportunities that these companies are losing every day because they are not extracting useful knowledge from their data. Sometimes, a timely contact to a customer given them the congregations for the increase in purchases or asking if there was something wrong because of the lake of purchases can really make the difference but useless if the contact is made 4 or 5 months later.

Forecasting creates probable future scenarios and anticipates problems that can be considered long before happening. But it is also important that these companies understand they are dealing with a machine learning algorithm that sometimes will fail. For example, there is no algorithm that can predict that a company's biggest competitor is about to bankrupt and that is selling the entire stock at half price. The way we dealt with this kind of uncertainty was by allowing us to detect and measure anomalies, at scale, when the observed values are outside of what had been predicted. In this way, we transform what could apparently be interpreted as a failure in the prediction algorithm into a feature that is alerting to something potentially important.

From a perspective more related to the development of a machine learning in an ERP for SMBs, we also found a set of challenges. First, there is the need for a strategy to deal with a high amount of time series. If we think that the number of a company's time series can easily reach thousands, and an ERP software producer also has thousands of customers, we easily reach billions of time series to forecast and detect anomalies. Our answer to this need can be shown in Figure 32 where we propose to divide the Forecasting and Anomaly detection into four main phases. The First one (Data Ingestion) the data in clean, stored and it is tested if the value is outside the predicted value. The send phase, Forecasting, is the most CPU consuming and can be done scheduled time or triggered by potential anomalies detected in the first phase. The third phase, Auto Optimize, is a scheduled task where a classifier is trained to select the best time-series forecast. Finally, there is a fourth phase where a quality report runs to output the accuracy of our system.

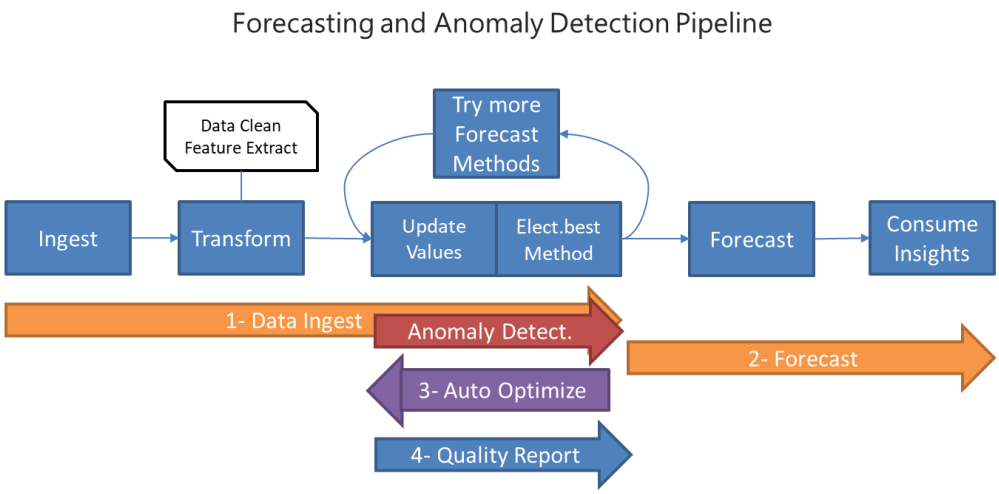


Figure 32 A proposed forecasting anomaly detection pipeline

Another challenge that we found was how to forecast and detect anomalies with companies' data that have no or small history. If we think of forecasting and anomaly detection as a feature in ERP software, a new customer expects to use this feature from the first month he is using the software. To deal with this problem, we propose "3 waves forecast" as shown in Figure 33. The "first wave", is a precise short-term forecast that will happen periodically as it is expected to change taking into account the most recent data. The "second wave" was called "increasingly precise long-term forecast" is following how close we are to meet companies' goals (like the end of the year goals). This is done by adding the actual value already fulfilled so far, and the expected value (forecast) until the end of the period. Finally, the third wave forecast is used to detect and measure anomalies, allowing us to generate alerts for potential problems or opportunities that may require attention

1. **Precise Short-Term Forecast**
   Objective: Short term goals
   Example: What is the expected income for this month?
   How: Monthly forecast 1 or 3 Month

Data	Real					F						
Month	1	2	3	4	5	6	7	8	9	10	11	12

2. **Increasingly Precise Long-Term Forecast**
   Objective: Define and Follow long term Business Goals
   Example: Follow how close to meet "total sales 2019" goal
   How: Monthly forecast to end of year; and calculate:
   "real total sales" + "forecast to end of year"

Goal	Total Sales for 2019											
Data	Real					Forecast						
Month	1	2	3	4	5	6	7	8	9	10	11	12

3. **Forecast for Detect Anomaly**
   Objective: Alert for abnormal behavior
   Example: Total sales = 0 for 2 weeks
   How: Calculate a target confidence interval and alert if outside that value (accumulator)

Alert Trg						R.Expected?						
Data	Real					Forecast						
Week	18	19	20	21	22	23	24	25	26	27	28	29
Month	5	5	5	5	5	6	6	6	6	7	7	7

Figure 33 Proposed "three-wave forecast" for a precise short-term, yearly increasing precise long-term, and for detect anomalies

The last challenge found, and present in any alert system is how to deal with a high amount of alerts. Our proposed solution is by measuring the anomalies with an accumulator that will deal with successive anomalies, but also will deal with error and then fixed issues, because they will have opposite signs, and finally will value newer anomalies over older anomalies. Using this feature, we reach a measure of the anomalies, in the same unit of the time series, and that will order all the time series that potentially require attention by the accumulated anomalies found, calculated on a given defined date. By using this anomaly score we are ordering anomalies and reducing false positives.

5.2. Research Questions Analysis

The key objective of this research was to be able to monitor time-series business metrics by learning the normal behavior, accurately predict future behavior and finally identify and score the abnormal behavior. We tried to answer the following research questions:

RQ1 What are the most common Machine Learning techniques for Autonomous Forecast of Business Metrics?

RQ2 What is the technical improvement of the proposed Autonomous Forecast Model versus a baseline?

RQ3 Can we detected deviations from expected behavior (anomalies) by using the prediction interval of the forecast model?

During the Related Work, we study the most common Machine Learning techniques for forecasting time series, that can be applied to Business Metrics. In Section 2.4 we studied a set of simple forecast methods summarized in Table 1. It was presented Exponential Smoothing and ARIMA methods, described as the most widely used approaches to time series forecasting. Finally, it was presented two most recent forecasting methods: Theta and Prophet. (RQ1) All these prediction modeling methods were implemented and evaluated during our investigation.

The Comparative Performance Model artifact allows us to conclude that the approach of trying to find a forecasting method that would perform well in any time series could not be the most suitable. For dealing with a wide diversity of time series we propose a meta-learning forecasting method that uses a classifier to identify the best forecasting method for each time series.

On our final approach, we worked with meta-learning with six forecasting methods, and we archived better predictive results than the baseline forecasting method in one, three and six-month forecasting periods. As summarized in Table 9, we were able to reduce the forecasting error using SMAPE from a mean of 122 in the baseline forecasting model to 69 in our proposed forecasting model (RQ2)

Finally, we were able to detect and measure deviations from de expected behavior (anomalies) using the prediction interval by proposing a “Pri Anomaly” metric that is an accumulator of the measure of local anomalies that allow us to highlight major accumulated anomaly time series. (RQ3)

5.3. Study Limitations

This investigation was supported by Primavera BSS. The main goal of Primavera was to develop a Time Series and Anomaly Detection Framework that could be consumed by all products in their portfolio. It was celebrated a protocol between Primavera BSS, ISCTE university and the student André Martins.

André Martins worked full time as an intern at the Lisbon delegation of Primavera BSS He was integrated in the "Innovation in New Technologies" department. Scrum methodology was used- It was defined sprinters that were reviewed monthly with all team members in Primavera headquarters in Braga. Daily Scrum meetings were made in the morning by video call to share the work done the day before, and the plan for the present day.

One of the biggest challenges that was considered in the key options, that have driven the present work, was the balance between the academic interest, and more practical interest of the Primavera BSS where the expectation of putting into production was assumed from the start.

Other limitations that were encountered during the investigation was the quality of the data. As it was assumed the intention to forecast and detect anomalies at scale, where the scale means the quantity and diversity of time series involved, we were naturally limited to the data that was provided and was already being used in the forecast baseline method that was the monthly profit and loss indicator of all the companies that used a specific cloud invoice software. It is mentioned that when working in a regression with monthly data we should have a theoretical minimum number of 14 observation, but it is also mentioned that "will be sufficient only when there is almost no randomness". This condition only applies to a very small number of companies and we could not force such a long historical data to provide predictions and anomaly detection.

A more technical limitation encountered was the lack of a good time series forecasting library in Python, as there is in R. This limitation was circumvented with a library that allows us to invoke R libraries in Python but naturally it took a lot more time as it involved two different programming languages and to use a complex library that exposes R objects to Python.

5.4. Future Research Proposition

In the present work we have proved that using the dataset shared by Primavera BSS, we were able to forecast and detect anomalies at scale

During the design and development phase, we prove that we couldn't find a single forecasting method with a good performance in all-time series provided, and in all forecasting periods. Prophet method that was developed for forecasting at scale outperformed the Primavera forecasting method used in production. Instead, a meta-learning classifier was built with a group of forecasting methods.

We suggest some future research that uses more forecasting methods and a more detailed study of the time series features. Another point that could lead to better results is to train a classifier that, instead of outputting the best forecasting method, worked with a weighted combination of the forecasting methods, that was proved in the M4 competition that could get excellent results. (Montero-Manso, et al., 2019)

In an ERP environment, the study of Hierarchical Forecasting would allow us to forecast series in different level of aggregation, and then reconcile the forecast results in order to remain consistent. An example of three levels of aggregation in time series is 1 – Sales of each Sales Customer; 2- Sales of each Sales Customer by Region; 3- Sales of all Customers.

The study of Dynamic Regression Models would allow us to include relevant information. " For example, the effects of holidays, competitor activity, changes in the law, the wider economy, or other external variables may explain some of the historical variation and may lead to more accurate forecasts" (Hyndman, et al., 2019) Chap. 9

Bibliography

Armstrong, J. Scott. 2001. Standards and Practices for Forecasting. *Principles of Forecasting: A Handbook for Researchers and Practitioners.* Norwell, MA : Kluwer Academic Publishers, 2001.

Bell, Franziska e Smyl, Slawek. 2018. Forecasting at Uber: An Introduction,. *Uber Engineering.* [Online] 6 de September de 2018. [Citação: 20 de January de 2019.] https://eng.uber.com/forecasting-introduction/.

Brown, R. G. 1959. *Statistical forecasting for inventory control.* s.l. : McGraw/Hill, 1959.

Brownlee, Jason. 2018. A Gentle Introduction to Exponential Smoothing for Time Series Forecasting in Python. [Online] 20 de August de 2018. [Citação: 2019 de January de 10.] https://machinelearningmastery.com/exponential-smoothing-for-time-series-forecasting-in-python/.

Brynjolfsson, Erik, Hitt, Lorin M. e Kim, Heekyung Hellen. 2011. *Strength in Numbers: How Does Data-Driven Decisionmaking Affect Firm Performance?* 2011.

Chandola, Varum, Banerjee, Arindam e Kumar, Vipin. 2009. *Anomaly Detection: A Survey.* s.l. : ACM Computing Surveys, 2009.

Chatfield, Chris. 2000. *Time-Series Forecasting.* Florida, USA : Chapman & Hall/CRC, 2000.

Colson, Eric. 2019. *What AI-Driven Decision Making Looks Like.* 8 de July de 2019. Havard Business Review.

Fulcher, Ben D. 2017. *Feature-based time-series analysis.* Melbourne, Australia : Monash Institute for Cognitive and Clinical Neurosciences, 2017.

Gartner. Gartner Glossary - Information Technology. *Gartner.* [Online] [Citação: 20 de July de 2019.] https://www.gartner.com/it-glossary/.

Goodwin, Paul. 2018. *Profit from Your Forecasting Software.* New Jersey : Wiley, 2018.

Hahmann, Martin. 2019. Big Data Analysis Techniques. [autor do livro] Sherif Sakr e Albert Zomaya. *Encyclopedia of Big Data Technologies.* Cham, Switzerland : Springer, 2019, p. 180.

Holt, C.E. 1957. *Forecasting seasonals and trends by exponentially weighted averages.* Carnegie Institute of Technology. Pittsburgh USA : s.n., 1957. O.N.R. Memorandum No. 52. https://doi.org/10.1016/j.ijforecast.2003.09.015.

Hyndman, Rob J. and Athanasopoulos, George. 2019. *Forecasting: Principles and Practice.* s.l. : OT, 2019.

Hyndman, Rob J. e Billah, Baki. 2001. *Unmasking the Theta method.* Melbourne : Department of Econometrics and Business Statistics, Monash University, 2001.

Hyndman, Rob J. 2014. Errors on percentage errors. *Rob J Hyndman.* [Online] 16 de April de 2014. [Citação: 01 de 06 de 2019.] https://robjhyndman.com/hyndsight/smape/.

Hyndman, Rob. 2018. Rob Hyndman - Feature-Based Time Series Analysis. *Youtube.* [Online] 21 de June de 2018. https://youtu.be/yx6OQ-8HofU.

klipfolio. Profit and Loss Report. *klipfolio.* [Online] [Citação: 15 de 7 de 2918.] https://www.klipfolio.com/resources/kpi-examples/financial/profit-loss-report.

LUCA, KLEINBERG, AND MULLAINATHAN. 2019. *ALGORITHMS NEED MANAGERS, TOO in HBT's 10 Must Reads On AI, Analytics and New Machine Age.* Boston, Massachusetts : HARVARD BUSINESS REVIEW PRESS, 2019.

Maximilian, Christ. tsfresh. *tsfresh.* [Online] Blue Yonder GmbH . [Citação: 01 de 05 de 2019.] https://tsfresh.readthedocs.io/en/latest/.

MCompetitions. M4 Competition Forecast. Compete. Excel. *M3.* [Online] [Citação: 14 de August de 2019.] https://www.mcompetitions.unic.ac.cy/m3/.

Mehrotra, Kishan G., Mohan, Chilukuri, Huang, Huaming. 2017. *Anomaly Detection Principles and Algorithms.* s.l. : Springer, 2017.

Montero-Manso, Pablo, et al. 2019. *FFORMA: Feature-based Forecast Model Averaging.* Victoria, Australia : Monash Business School, 2019.

Ord, Keith, Fildes, Robert and Kourentzes, Nikolaos. 2017. *Principles of Business Forecasting 2nd Edition.* New York : Wessex Press, Inc., 2017.

Pal, Avishek e Prakash, PKS. 2017. *Practical Time Series Analysis; Master Time Series Data Processing, Visualization, and Modeling using Python.* Birmingham : Packt Publishing Ltd., 2017.

Peffers, Ken, et al. 2008. *A design science research methodology for information systems research.* 2008. pp. 45-77.

Provost, Foster e Fawcett, Tom. 2013. *Data Science for Business.* Sebastopol, California : O'Reilly Media, 2013.

Real-time anomaly detection system for time series at scale. **Toledano, Meir, et al. 2017.** Halifax, Nova Scotia - Canada : s.n., 2017. KDD 2017: Workshop on Anomaly Detection in Finance.

Sakr, Sherif e Albert, Zomaya. 2019. *Encyclopedia of Big Data Technologies.* Cham, Switzerland : Spriger, 2019.

Shipmon, Dominique, et al. 2017. *Time Series Anomaly Detection; Detection of Anomalous Drops with Limited Features and Sparse Examples in Noisy.* Cambridge, Massachusetts, USA : Google, Inc., 2017.

Talagala, Thiyanga, Hyndman, Rob e Athanasopoulos, George. 2019. *Meta-learning how to forecast time series.* Melbournehideman : Monash Business School, 2019.

Taylor, Sean J. and Letham, Ben. 2017_2. Facebook research; Prophet: forecasting at scale. [Online] February 23, 2017_2. [Cited: January 2019, 10.] https://research.fb.com/prophet-forecasting-at-scale/.

Taylor, Sean J. e Letham, Benjamin. 2017. Forecasting at Scale. *PeerJ — the Journal of Life and Environmental Sciences.* [Online] 2017 de September de 2017. https://doi.org/10.7287/peerj.preprints.3190v2.

Winters, P. R. 1960. *Forecasting sales by exponentially weighted moving averages.* 1960. pp. 324–342, Management Science. https://doi.org/10.1287/mnsc.6.3.324.

Zhu, Lingxue e Laptev, Nikolay. 2017. *Deep and Confident Prediction for Time Series at Uber.* New Orleans, LA, USA, : s.n., 2017.

A Annex and Appendix

Artificial intelligence (AI)

"Artificial intelligence AI applies advanced analysis and logic-based techniques, including machine learning, to interpret events, support and automate decisions, and take actions." From https://www.gartner.com/it-glossary/artificial-intelligence/

Balanced Scorecard (BSC)

"A balanced scorecard (BSC) is a performance measurement and management approach that recognizes that financial measures by themselves are not sufficient and that an enterprise needs a more holistic, balanced set of measures which reflects the different drivers that contribute to superior performance and the achievement of the enterprise's strategic goals. The balanced scorecard is driven by the premise that there is a cause-and-effect link between learning, internal efficiencies and business processes, customers, and financial results." From https://blogs.gartner.com/it-glossary/balanced-scorecard/

Big data

"Big data is high-volume, high-velocity and/or high-variety information assets that demand cost-effective, innovative forms of information processing that enable enhanced insight, decision making, and process automation." From https://www.gartner.com/it-glossary/big-data/

Business intelligence (BI)

"Business intelligence is an umbrella term that includes the applications, infrastructure and tools, and best practices that enable access to and analysis of information to improve and optimize decisions and performance." From https://www.gartner.com/it-glossary/business-intelligence-bi/

Business intelligence (BI) platforms

"Business intelligence platforms enable enterprises to build BI applications by providing capabilities in three categories: analysis, such as online analytical processing (OLAP); information delivery, such as reports and dashboards; and platform integration, such as

BI metadata management and a development environment." From https://blogs.gartner.com/it-glossary/bi-platforms/

Customer relationship management (CRM)

"Customer relationship management is a business strategy that optimizes revenue and profitability while promoting customer satisfaction and loyalty. CRM technologies enable strategy, and identify and manage customer relationships, in person or virtually. CRM software provides functionality to companies in four segments: sales, marketing, customer service and digital commerce." From https://blogs.gartner.com/it-glossary/customer-relationship-management-crm/

Dashboards

"Dashboards are a reporting mechanism that aggregate and display metrics and key performance indicators (KPIs), enabling them to be examined at a glance by all manner of users before further exploration via additional business analytics (BA) tools. Dashboards help improve decision making by revealing and communicating in-context insight into business performance, displaying KPIs or business metrics using intuitive visualization, including charts, dials, gauges and "traffic lights" that indicate the progress of KPIs toward defined targets." From https://blogs.gartner.com/it-glossary/dashboard/

Data Lake

"A data lake is a collection of storage instances of various data assets additional to the originating data sources. These assets are stored in a near-exact, or even exact, copy of the source format. The purpose of a data lake is to present an unrefined view of data to only the most highly skilled analysts, to help them explore their data refinement and analysis techniques independent of any of the system-of-record compromises that may exist in a traditional analytic data store (such as a data mart or data warehouse)." From https://blogs.gartner.com/it-glossary/data-lake/

Data Mining

"The process of discovering meaningful correlations, patterns and trends by sifting through large amounts of data stored in repositories. Data mining employs pattern

recognition technologies, as well as statistical and mathematical techniques." From https://www.gartner.com/it-glossary/data-mining

Data Warehouse

"A data warehouse is a storage architecture designed to hold data extracted from transaction systems, operational data stores and external sources. The warehouse then combines that data in an aggregate, summary form suitable for enterprise-wide data analysis and reporting for predefined business needs.

The five components of a data warehouse are:

1. production data sources
2. data extraction and conversion
3. the data warehouse database management system
4. data warehouse administration
5. business intelligence (BI) tools

A data warehouse contains data arranged into abstracted subject areas with time-variant versions of the same records, with an appropriate level of data grain or detail to make it useful across two or more different types of analyses most often deployed with tendencies to third normal form. A data mart contains similarly time-variant and subject-oriented data, but with relationships implying dimensional use of data wherein facts are distinctly separate from dimension data, thus making them more appropriate for single categories of analysis." From https://blogs.gartner.com/it-glossary/data-warehouse/

Enterprise resource planning (ERP)

"Enterprise resource planning is defined as the ability to deliver an integrated suite of business applications. ERP tools share a common process and data model, covering broad and deep operational end-to-end processes, such as those found in finance, HR, distribution, manufacturing, service and the supply chain.

ERP applications automate and support a range of administrative and operational business processes across multiple industries, including line of business, customerfacing, administrative and the asset management aspects of an enterprise. However, ERP deployments tend to come at a significant price, and the business benefits are difficult to justify and understand.

Look for business benefits in four areas: IT cost savings, business process efficiency, as a business process platform for process standardization and as a catalyst for business

innovation. Most enterprises focus on the first two areas, because they are the easiest to quantify; however, the latter two areas often have the most significant impact on the enterprise. In 2013, Gartner defined the Postmodern ERP environment." From https://blogs.gartner.com/it-glossary/enterprise-resource-planning-erp/

Machine Learning

"Advanced machine learning algorithms are composed of many technologies (such as deep learning, neural networks and natural-language processing), used in unsupervised and supervised learning, that operate guided by lessons from existing information." From https://www.gartner.com/it-glossary/machine-learning/

Neural Net or Neural Network

"A neural net or neural network is an artificial-intelligence processing method within a computer that allows self-learning from experience. Neural nets can develop conclusions from a complex and seemingly unrelated set of information." From https://blogs.gartner.com/it-glossary/neural-net-or-neural-network/

Natural-language processing (NLP)

"Natural-language processing technology involves the ability to turn text or audio speech into encoded, structured information, based on an appropriate ontology. The structured data may be used simply to classify a document, as in "this report describes a laparoscopic cholecystectomy," or it may be used to identify findings, procedures, medications, allergies and participants." From https://blogs.gartner.com/it-glossary/natural-language-processing-nlp/

Predictive Behavior Analysis

"The use of techniques such as data mining, data visualization, algorithm clustering, and neural networking to find patterns or trends in data. These patterns or trends are used to forecast future behavior based on current or past behavior. Uses of predictive behavior analysis include identifying customers likely to drop out or default; identifying products customers are likely to buy next; developing customer segments or groups; and product development." From https://blogs.gartner.com/it-glossary/predictive-behavior-analysis/

Predictive modeling

"Predictive modeling is a commonly used statistical technique to predict future behavior. Predictive modeling solutions are a form of data-mining technology that works by analyzing historical and current data and generating a model to help predict future outcomes. In predictive modeling, data is collected, a statistical model is formulated, predictions are made, and the model is validated (or revised) as additional data becomes available. For example, risk models can be created to combine member information in complex ways with demographic and lifestyle information from external sources to improve underwriting accuracy. Predictive models analyze past performance to assess how likely a customer is to exhibit a specific behavior in the future. This category also encompasses models that seek out subtle data patterns to answer questions about customer performance, such as fraud detection models. Predictive models often perform calculations during live transactions—for example, to evaluate the risk or opportunity of a given customer or transaction to guide a decision. If health insurers could accurately predict secular trends (for example, utilization), premiums would be set appropriately, profit targets would be met with more consistency, and health insurers would be more competitive in the marketplace." From https://blogs.gartner.com/it-glossary/predictive-modeling/

Postmodern ERP

"Postmodern ERP is a technology strategy that automates and links administrative and operational business capabilities (such as finance, HR, purchasing, manufacturing and distribution) with appropriate levels of integration that balance the benefits of vendor-delivered integration against business flexibility and agility. This definition highlights that there are two categories of ERP strategy: administrative and operational.

Administrative ERP Strategy. This focuses on the administrative aspects of ERP, primarily financials, human capital management and indirect procurement. Some industries don't need operational capabilities, such as manufacturing or distribution, so they focus their ERP strategy on administrative functions, perhaps augmented by some industry-specific functionality (such as grant management in the higher education and public sectors, or project resourcing, billing and costing in professional services). These industries are generally characterized as service-centric industries.

Operational ERP Strategy. Organizations in manufacturing, distribution, retail, etc. (sometimes referred to as product-centric industries) are likely to extend their ERP strategy beyond administrative functions into operational areas, such as order management, manufacturing and supply chain, to maximize operational efficiencies. Also, asset-intensive organizations, such as utilities and mining, may include operations and maintenance of assets in their ERP strategy. These organizations can realize benefits from the integration between administrative and operational capabilities, for example, where operational transactions that have a financial impact are reflected directly in the financial modules." From https://blogs.gartner.com/it-glossary/postmodern-erp/

Predictive modeling

"Predictive modeling is the process of analyzing data to create a statistical model of future behavior. Predictive modeling solutions are a form of data-mining technologies that work by analyzing historical and current data, and generating a model to help predict future outcomes. These technologies can be used to generate a score (for example, a credit score), to assess behavior (for example, fraud detection or customer acquisition), or to analyze needed reserves. Insurers can apply this to key activities, such as customer service, pricing, actuarial, underwriting and claims, to improve outcomes."
From https://blogs.gartner.com/it-glossary/predictive-modeling-solutions/

RDBMS (relational database management system)

"A database management system (DBMS) that incorporates the relational-data model, normally including a Structured Query Language (SQL) application programming interface. It is a DBMS in which the database is organized and accessed according to the relationships between data items. In a relational database, relationships between data items are expressed by means of tables. Interdependencies among these tables are expressed by data values rather than by pointers. This allows a high degree of data independence." From https://blogs.gartner.com/it-glossary/rdbms-relational-database-management-system/

Real-time analytics

"Real-time analytics is the discipline that applies logic and mathematics to data to provide insights for making better decisions quickly. For some use cases, real time simply means the analytics is completed within a few seconds or minutes after the arrival of new data. On-demand real-time analytics waits for users or systems to request a query and then delivers the analytic results. Continuous real-time analytics is more proactive and alerts users or triggers responses as events happen." From https://blogs.gartner.com/it-glossary/real-time-analytics/

Small and midsize business (SMB)

"A small and midsize business is a business which, due to its size, has different IT requirements—and often faces different IT challenges—than do large enterprises, and whose IT resources (usually budget and staff) are often highly constrained. For the purposes of its research, Gartner defines SMBs by the number of employees and annual revenue they have. The attribute used most often is number of employees; small businesses are usually defined as organizations with fewer than 100 employees; midsize enterprises are those organizations with 100 to 999 employees. The second most popular attribute used to define the SMB market is annual revenue: small business is usually defined as organizations with less than $50 million in annual revenue; midsize enterprise is defined as organizations that make more than $50 million, but less than $1 billion in annual revenue." From https://www.gartner.com/it-glossary/smbs-small-and-midsize-businesses

Software as a Service (SaaS)

"Gartner defines software as a service (SaaS) as software that is owned, delivered and managed remotely by one or more providers. The provider delivers software based on one set of common code and data definitions that is consumed in a one-to-many model by all contracted customers at anytime on a pay-for-use basis or as a subscription based on use metrics." From https://blogs.gartner.com/it-glossary/software-as-a-service-saas/

Visualization

"Visualization is the illustration of information objects and their relationships on a display. Strategic visualization graphically illustrates the strength of relationships by the proximity of objects on the display. Advanced technology can make a significant difference in users' ability to interface to large knowledge repositories. These advances

use the distance between objects on the display to reflect the similarity of meaning, similarity of content or other relationships (e.g., association with a group).”From https://blogs.gartner.com/it-glossary/visualization/