Design of monitoring applications and prediction of key industrial metrics: IIoT + AI

Master Thesis
submitted to the Faculty of the
Facultat d'Informàtica de Barcelona
Universitat Politècnica de Catalunya
by

Pablo Pazos Domínguez

In partial fulfillment
of the requirements for the
Master in Innovation and Research in Informatics – Data Science

Advisor: Daniel Méndez Martí
Tutor: Miquel Sànchez-Marrè
Barcelona, June 2022

Abstract.- Global industry has suffered deep changes in last years because of the successful development and integration of new technologies. Industry 4.0 has emerged as a new standard for achieving efficiency and improve processes. Among the technologies used in Industry 4.0, Internet of Things applied to industry (IIoT) enable real-time, intelligent, and autonomous access, collection, analysis, communications, and exchange of process, product and/or service information, within the industrial environment, so as to optimize overall production value. Because of its importance, in this project, a methodology for extracting, analyzing and using the data gathered by IIoT devices is proposed in order to extract meaningful information and to predict industrial key metrics with Artificial Intelligence. In addition, for the complete validation of the proposed methodology, a practical implementation of all the mentioned aspects is carried out by developing a study of the industrial process in the wastewater treatment field using the data collected by an Industrial Internet of Things infrastructure and modeling key time series metrics, such as total organic carbon (TOC) and carbon removal performance (CRP) by using Machine Learning models XGBOOST Regressor, Multi-Layer Perceptron (MLP) Regressor and Support Vector Regressor (SVR) to implement a dashboard with an operational panel and a decision-making panel that help anticipate possible deviations in the performance of the industrial process.

keywords.- Industrial Internet of Things methodology, time series analysis, time series prediction, time series visualization

Contents

List of Figures

List of Tables

1 Introduction

1.1 Context

Global industrial landscape has changed over the last few years in a radical manner due to the innovations in technology development. Industry 4.0 has become the solution to key problems related with performance and efficiency. This new paradigm has deeply change how industry works and it brings consequences on processes, industry, economy and markets.

Industry 4.0 brings to the scenario new participants like Cyber-Physical Systems, Internet of Services (IoS), Internet of Things (IoT), Big Data, Analytics, Robotics and Cloud Manufacturing. Those new technologies allow all the participants to exchange information between them, triggering actions, to control each other, to collect large amount of information and to create an intelligent industrial environment. This new approach will bring improvement in productivity rates and efficiency among companies that adopt this new paradigm. Not only IIoT based systems can adopt easily new regulations and tight the process to the existing ones but they will perform better regarding time-efficiency and performing rates.

With the introduction of Internet of Things in industry and the exchange of information between devices, the amount of data collected has become an important aspect for companies. The study and analysis of data collected and exchanged between devices has given companies the possibility of understand deeply the key factors of the industrial process and, by using Artificial Intelligence, to predict important metrics for their business.

As Industry 4.0 is becoming the standard, new methodologies have to be applied to create data-driven systems that allow companies to extract information of industrial process and to proactively predict industrial key metrics. In this thesis, we present a new methodology of designing a monitoring and predicting application for industrial key metrics by using information collected by Industrial Internet of Things and a data-driven model able to give meaningful information. The mentioned methodology will be proven by creating a system under the use case of wastewater treatment plant by using the process parameters collected with Industrial Internet of Things (IIoT) devices, consolidating that information, creating a data-driven model able to predict the key metrics of the process: total organic carbon value (TOC) and carbon removal performance (CRP) and presenting its insights to the final user by creating a visualization application. This use case will cover all the phases of the methodology proposed and will help to assess its correctness and validity.

1.2 Motivation

This Master Thesis is result of my work in the company IThinkUPC located in Barcelona under the tutorship of Daniel Méndez Martí. There I have been working on a project in the Industrial Internet of Things field. By working building an intelligent solution for a wastewater treatment company, I have found the need of an intelligent tool that help plant operators to anticipate the values of TOC and CRP to be able to proactively react to future up-trends in those plant parameters.

Project origin was the identification of the need for an intelligent solution that exploit data gathered with Industrial Internet of Thing devices to characterize their industrial of water treatment process by giving insights about key metrics performance such us value of organic carbon degradation (TOC) and the carbon removal performance (CRP) and, then, with use of prediction, to act proactively in order to avoid metric deviations and ensure the correct behaviour of the plant processes. Monitoring of those metrics is vital for the correct operation of the plant, based on their values, operators have to modify process parameters so as to ensure that TOC and CRP are correctly in range regarding the strict regulations.

The final use case is included in the project to complement the importance of creating a general methodology that captures all the steps needed to create the mentioned system. This methodology has to move the use case-driven process to a data-driven one and has to lead the theoretical development of an intelligent data-driven system with the ability of monitoring and prediction key industrial metrics by using data collected in the process with IIoT devices. It has to be completely decoupled from the use case and satisfy general objectives for all, complying with the theoretical design requirements. Once the objectives have been achieved, practical case will be designed and developed under the methodology proposed.

1.3 Objectives and thesis structure

1.3.1 Main objective

Having into account the Industry 4.0 challenges and the problematic stated in the Context and Motivation sections, the main objective of this project is to formalize the methodology of how to design and create a monitoring and predicting application for industrial key metrics by using information gathered by Industrial Internet of Thing devices and to create a data-driven model able to give meaning information based on that data. In an ideal scenario, the final objective of the thesis would be to obtain a methodology that can completely decouple from the practical use case indented to be developed. Because of completely decoupling is hard to obtain, the system to be implemented has to have some characteristics such us parameter and data gathering using IIoT devices, the use of an artificial intelligence model and the visualization of those insights reached by the model.

In order to achieve it, we have to stablish the three core parts of the architecture (infrastructure, model and visualization), their importance, their functionalities and their contribution to the correct performance of the system.

Combined with the statement of the methodology, a real system implementation will be developed to serve the monitoring and prediction of TOC and CRP signals in the wastewater treatment field. The real system will be asses the validity and the correctness of the methodology.

1.3.2 Specific objectives

The first specific objective is to explore the Industrial Internet of Things literature proposed up to today to capture objectives, limitations, advantages and disadvantages of using this kind of technology in industrial data gathering and process monitoring.

The second specific objective that we have is to design the complete system architecture, based on Industrial Internet of Thing devices and to fully exploit the data collected and exchanged between them. This objective ranges from the location of the IIoT devices to the consolidation of the data collected by using and extraction, transformation and loading processes that will store the information in a database, ready to be analysed and, later, exploited by the machine learning model.

Once we have the data consolidated, the third specific goal is to establish how an artificial intelligence model has to use the data in order to achieve meaningful insights in key industrial metric prediction by stating vital steps to obtain validity and correctness. Those steps have to be as much independent as possible from the use case, having into account some assumptions that will be written later in this memory. This analysis and model development will be based in signal analysis and auxiliary signal analysis and it also includes the complete machine learning workflow composed by the exploratory data analysis (EDA), model selection, model training, model evaluation and deployment.

In particular, as it was mentioned before, industrial metric forecasting is one of the important objectives to achieve and it is the main task of the model phase. In order to achieve this goal, this project will review and use, the models proposed in the up-to-date literature, including industrial process monitoring by means of neural networks (ANN), support vector regression (SVR) for industrial process monitoring, time-series XGBOOST regression, within others.

In terms of the final user experience, the fourth specific goal is to obtain a user-friendly application in which all the information obtained by the model can be displayed clearly. This application consumes the information consolidated with the infrastructure, the forecasting and the meaningful insights achieved by the model developed.

In order to achieve those goals, the thesis will also review the updated tools for each one of the phases of the real implementation of the methodology and will present their pros and cons.

1.3.3 Thesis structure

In section Background and state of art relevant concepts for the development of this project are going to be covered. Industry 4.0 and Industrial Internet of Things will cover the Industry 4.0 and the basis of Industrial Internet of Things and a general architecture. In IIoT and Data Science, works that combine both IIoT and Data Science are going to be explained including their approaches. Regarding the practical use case, in Wastewater treatment the specific problematics of this field and the solutions of other authors are going to be reviewed.

In Theoretical Methodology, the proposed framework for Industry 4.0 and the industrial field that uses Data Science to achieve important insights about industrial processes is going to be analysed. In section CRISP-DM Methodology, the IIoT CRISP-DM Methodology for data science projects is explained and modified to satisfy the necessities of the introduction of the Internet of Things, by including personalized steps such us IoT architecture design and industrial signal analysis.

Regarding the real use of the methodology, in Practical use of the methodology the know-how proposed in the theoretical methodology is used to address the wastewater treatment industrial problem. Trying to extract meaning information about the industrial process and to achieve a model which is able to predict chaotic key industrial metrics which define the correct performance of the plant. Theoretical methodology is going to guide the complete development of the system and will assess if the methodology was well design for the industrial world.

2 Background and state of art

In this section, important papers related with the research in the most relevant areas to the thesis are going to be reviewed.

Firstly, in Industry 4.0 and Industrial Internet of Things. we cover the most important aspect regarding Industry 4.0 and Industrial Internet of Things. This includes the fundamentals, why the industry has evolved towards industry 4.0, why it is important and which advantages does it bring to the industry field.

Secondly, in IIoT and Data Science. we cover IIoT in combination with Data Science. Data Science is a very broad topic and there are many fields of study, but we are only covering the researches related to this thesis. This includes information about the complete data workflow, from how data gathered by IIoT is extracted, transformed and loaded into a database to visualization of metrics acquired by an artificial intelligence model, going through Industrial process monitoring and prediction of key industrial metrics with artificial intelligence.

Lastly, regarding the practical use case, in Wastewater treatment literature regarding this topic is reviewed. This encloses the distribution of a wastewater treatment plant and its main parts, also how and why IIoT devices are used to gather data and monitor the plant's parameters, which are the important metrics and signal to be monitored and predicted and why they are important to the use case.

2.1 Industry 4.0 and Industrial Internet of Things

2.1.1 Industry 4.0

Industry 4.0 was firstly introduced in Germany in 2011 [1], once the idea was evolving, Germany officially adopted it in 2013 as a German strategic initiative to take a step forward in the modernization and updating of the industrial network of the country. The introduction of Industry 4.0 establishes the beginning of the fourth industrial revolution, which is based in the introduction of technologies such as cyber-physical systems (CPS), the Internet of Things (IoT), the Internet of Services (IoS) and Cloud Computing. In Industry 4.0 paradigm, embedded systems, machine-to-machine communication (M2M), IoT, IoS and CPS technologies are introducing the digital space to the physical one and, with that, new generation of industrial actors, such us smart factories, are appearing to deal with the increasing complexity and new challenges faced by the industrial field. Industry 4.0 is focused on bringing end-to-end digitization and the introduction of smart and digital industrial ecosystems by creating complete integrated solutions [2].

In Industry 4.0, IoT offers transformational solutions for the operation and functioning of many existing industrial systems [3]. It has revolutionized the existing manufacturing systems in production so, it is considered a key participant in the new generation of smart industry. IoT gives the fundamental concept of integration of all devices that are part of the industrial process by enabling creation of virtual networks to support the gathering and exchange of information between all the participants in the network. The introduction of IoT provides new opportunities for users, manufactures and companies as it changes the industry to become smarter, autonomous, more reliable and provide added-value services and products. It has a great impact in several industries such us automation, industrial manufacturing, business processes, logistics and transportation. It provides three main applications: (1) process optimization, (2) optimized resource consumption and (3) creation of complex autonomous systems.

Internet of Services (IoS) [3] have emerged recently and it brings new opportunities to the service industry. It provides the capability of building business networks between service providers and customers based on technology. It has a similar approach to IoT, but, as its name suggests, it is applied to services instead of physical entities. IoS can be summarized as a new business model that changes deeply the way that services are provided to the user by increasing the value of creation gathered from all the participants in the industry environment, such us the organization, the customers, suppliers and intermediators.

Cloud computing contributes significantly to the realization of the Industry 4.0. It uses a network of resources by distributing them within the network. It is widely used in Manufacturing-as-a-Services (MaaS) and it has been gaining attraction in the last few years.

Cyber-physical systems are the core of the introduction of Industry 4.0 [3]. It has been one of the most important developments in information and communication technology. They are systems made of collaborative entities and connected to the physical surroundings and their processes and they create an intelligent network and a smart production. In order to achieve that goal, CPS use data-accessing and data-processing services available on the Internet. CPS have physical and software elements that are under high coordination standards because they have evolved from embedded systems. Last researches confirmed that the introduction of CPS has helped factories to create a communication environment between machines and decentralized control over their systems that lead to an production optimization. The introduction of CPS in the industry allows vertical and horizontal integration with information technology systems and the interconnection of the whole industry workflow, potentially transforming the today factories and industries by introducing intelligence in the manufacturing process.

Industry 4.0 brings with it a complete set of new technologies and it completely change the manner of how industrial processes work till now. The digital transformation advancements and the interconnection between all the participants bring new challenges to the industry, and they change several domains beyond the industry sector, new impacts can be categorized in 6 areas: (1) Industry, (2) Products and services, (3) Business models and market, (4) Economy, (5) Work environment and (6) Skills development.

In conclusion, Industry 4.0 brings the potential of transforming the industry by introducing new technologies such as IoT, IoS, Cloud Computing and CPS. It can be said that it will change completely the paradigm aborded now in the manufacturing process and it provides improvements in the production, engineering processes by enhancing the quality of products and services, bringing new business opportunities and interconnecting all the participants in the industrial workflow.

2.1.2 Industrial Internet of Things

It has been mentioned the importance of Internet of Things in the industry sector by becoming a key participant in the complete transformation of the industry landscape. It has become so important that new researches have appeared combining both fields, this new study sector is named Industrial Internet of Things (IIoT). IIoT brings a new vision of IoT in the industrial sector by introducing smart devices for sensing, gathering, processing and communicating the real-time processes or events in the industrial systems. W.Z. Khan, M.H. Rehman *et al.* [6] define the IIoT as a network of intelligent and highly connected industrial components that are deployed to achieve high production rate with reduced operational costs through real-time monitoring, efficient management and controlling of industrial processes, assets and operational time.

Although IIoT can be considered a subset of IoT, they have differences between them. IIoT requires higher level of safety, security and reliability in communication without the distortion of real-time communication in industrial operations due to its critical mission in industrial environments. It has also a different objective that is the efficient management of industrial assets such as machines, industrial devices and operations including predictive maintenance.

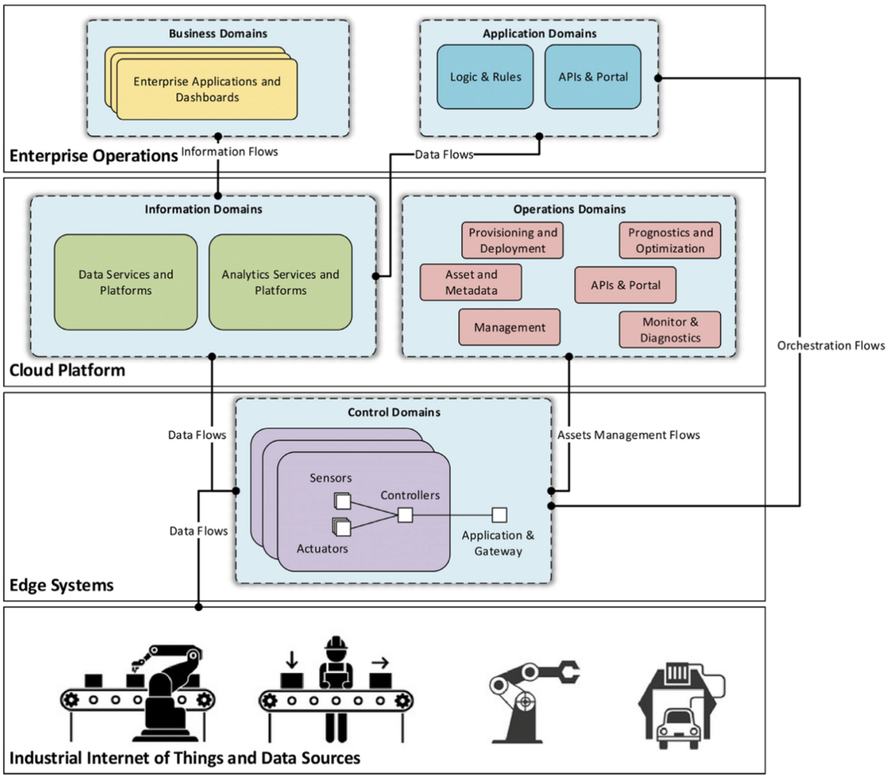


Figure 1: A generical schema for IIoT systems in which it can be seen the different parts of the IIoT structure. [5]

- Infrastructure architectures in IIoT: han, Wazir Zada and Rehman. MH *el al.* [5] presented this generic architecture. As we can see in A schema architecture for IIoT systems, architecture is divided in 4 layers. IIoT devices and industrial data sources generate data streams by gathering and communicating information between them at Layer 1, whereas the edge servers and cloud computing systems complement it at Layer 2 & 3 by empowering the system. Enterprise operations are presented in Layer 4. This figure presents an overview of how data flows among different layers and the orchestration of all the participants to achieve resource management in the industrial networks. However, there are not a full consensus between all the researchers and there are different interpretations of how an ideal architecture for IIoT system would be. There are different considerations regarding location awareness, communication paradigms, computational assignments, execution paradigms, resource management schemes, safety, security, privacy, addressability, and resilience, within others.

- Communication protocols for IoT: There is no particular communication protocol in IIoT. As many authors suggest, there are several communication protocols that vary from each other in topology, latency and hardware-software. To mention some of them, Meng *et al.* [18] proposed a ZMQ messaging protocol based on a distributed topology and high latency by using Matlab and Visual C. Katsikeas *et al.* [19] proposed a protocol with MQTT 3.1.1., OASIS, ISO/IEC, TLS, IPsec technologies with a distributed star topology and giving optimized high latency between other studies.

- Data management in IIoT: Once again, there are several approaches to achieve this goal. Some of them are one proposed by Theofanis *et al.* [20] based on a distributed Data Management Layer (DML) for storage data in IIoT that interact with the network layer to help in identifying the network nodes for generating, storing and requesting the data. Lucas-Estan *et al.* [21] proposed a software approach based on a hierarchical and multi-tier architecture for network connectivity and data management, it enables data distribution and network connectivity through different available heterogeneous and unlicensed wireless connections, such as 5G, between others.

The main importance of IIoT is that it allows the establishment of an infrastructure on which a big set of technologies can be deployed, for instance, IoT, cloud computing, big data analytics, artificial intelligence, cyber-physical systems, augmented reality, virtual reality, Humane-to-Machine (H2M), and M2M communication. Those technologies can fully exploit the new possibilities brought by large amount of data gathered by the IIoT backbone that can be monetized and it improve the overall performance of the systems for providing new services.

2.2 IIoT and Data Science

Due to the fact that IIoT devices constantly monitor their environment, they generate a massive amount of data that can be used by highly sophisticated high performance computer systems for data analysis or systems that run autonomously with intelligence whose objective is to minimize the human-interaction, improve efficacy and/or create meaningful insights about the state of the industrial processes to the final user. Data scientists have been exploring the creation of big data applications with well-designed data science methods to analyze and model big volumes of structured and unstructured data.

The main goal of those systems is to extract information that allows identifying trends, discovering correlations, predicting patters and undertaking complex but effective decisions. Once applied to industry, those systems are able to offer enhanced and deep data collection from smart machines, real-time system monitoring, increasing efficiency and productivity.

Sarfraz Nawaz Brohi, Mohsen Marjani *et al.* [7] IIoT as the telemetry analysis IoT based that can be used in the industrial field, we have decided to evolve this approach by going deeply in the theoretical implementation and by stating the procedure to achieve a good system with the premises introduced in Objectives and thesis structure. As seen in the state of art, Sarfraz Nawaz Brohi, Mohsen Marjani *et al.* [7] introduced a methodology composed by three main parts: (1) plan (2) collect and (3) analytics. Phases of this approach are described in detail:

- Plan: This phase is related with the acquisition of the project requirements in which all the project stakeholders must be involved in the planning to ensure that their requirements are correctly understood and analysed. In addition, all the domain experts have to be involved in every project cycle to provide domain knowledge and review the continuous advances of the system as well as the direction of the solutions to perceive valuable insights and the required information. For the authors, after this first successful requirements gathering, the data scientist can start modulating its analysis by using statistical techniques and machine learning. After theses preliminary finding, both technical stakeholders and data scientist can start working together to select the most suitable methods to use and the selection of local, hybrid or cloud infrastructure. In this phase, it is very important to analyze the IoT data sources because telemetry from unknown or unreliable data sources can lead to error in the analysis phase.
- Collect: This phase starts after the correct ending of the plan phase. The authors stablish the use of a gateway as the communication mechanism between the IoT hub and the IoT data sources for controlling the data obtained. This gateway manages all IoT opened connections and implements semantics for multiple protocols such us MQTT, CoAP, WebSockets of HTTP, to ensure that the successful communication is performed. In addition, the gateway can perform SQL-like instructions for filtering and selecting the incoming data. This gateway publishes all the data that can be consumed by downstream analytical systems using stream or batch processing.
- Analytics: Data science analysis would be applied over blocks with data generated within a period of time and that have been extracted by using batch-processing over the telemetry data. Although this analysis is static, real-time analysis can be done as the connection and gateways skills allows the analytics system to obtain the data required in each moment. Each analysis could be useful in different situations, such as analysis of data from financial services over a week for static analysis and fraud prevention for real-time analysis. After applying batch or streaming processing, data need to be cleaned in a pre-process step by removing duplicated samples, removing

null values and handling outliers, this is done in the cleaning phase. Unlike manual data cleaning, in IoT environments, automation of data workflows is very desirable for avoid ad-hoc behaviors. Then, prepared data is analyzed by using exploratory data analysis methods and used to feed machine learning and statistical models. To finish, visualization of the insights achieved by the models are displayed to the final user.

Although the PCA-IoT methodology is a good approach, it is described at high level and written with few technical details very important for achieving a practical implementation from a framework. The plan, collect and analytics phases are very general and each of them can be divided into work packages that are important enough to create subtasks in themselves. Also, PCA-IoT does not fulfill all the goals stablished in Objectives and thesis structure as it does not stablish a general methodology for industrial signal analysis and prediction with IoT by giving importance to industrial signal monitoring. For all those reasons, a new theoretical methodology is created by using the main ideas stablished Sarfraz Nawaz Brohi, Mohsen Marjani *et al.* [7] but evolving the idea until achieving a completely functional framework that support the creation of a fully capable system whose objective is the signal monitoring and prediction in IIoT.

Michael Horrell, Larry Reynolds, and Adam McElhinney [23] focus their studies in the heavy industry, more specifically in farming industry analytics problems in equipment reliability and predictive maintenance. They proposed and top-down approach for creating value from industrial IoT data. The first incise in how logics behind industry field model the KPI's and industrial metrics to be analyzed. They state that the core of solving a predictive maintenance problem involves gathering data, conducting analysis, building and deploying a model, and tracking outcomes and feedback to ensure the model is performing appropriately.

2.3 Industrial Process Monitoring

Reliability, safety and robustness are critical factors in modern industry and, with new introductions of technology to achieve Industry 4.0 paradigm, a lot of effort have been done to integrate process monitoring and industrial metric analysis during the last years. Being able to predict the future of condition of industrial processes is becoming more and more important to achieve efficiency, efficacy and reliability in the industrial process outcome. In order to achieve that goal, industrial process monitoring has become a key participant in the industry ecosystem of today's world and researchers have been addressing this problem in their last studies. Researchers also focus their studies in finding suitable models that model future behaviors of critical industrial signals.

For Daniel Zurita, Miguel Delgado et al. [10] there are two main challenges to be addressed: (1) the consideration of suitable procedures to deal with highly non-linear signal behaviours where the correlation of objective signal with the rest of process information quickly decreases within a short period of time. (2) The assessment and exploitation of such auxiliaryy information related with the objective signal, which is required to enhance the forecasting performance avoiding computational complexity and model overfitting, being this tradeoff between complexity and overfitting one of the key aspects to have in mind.

The prediction of future signal behavior can be seen as a time series prediction problem. Although, there is a significative amount of literature in time series analysis, signal prediction has particular challenges and particular methods need to be applied in order to achieve meaningful results.

For Instance, Su *et al.* [8] proposed an Adaptative Neuro Fuzzy Inference System (ANFIS) to predict the evolution of a non-linear time series. To obtain input features, they use a non-linear input selection model based on Adaptive Selection Method. The ANFIS model combine the parametric adaptability of neural networks and the generalization capabilities of fuzzy logics and, therefore, ANFIS forecasting gives a very reliable and robust condition predictor because it can capture non-linear input reactions accurately. Because of their characteristics. Although, ANFIS model is widely used in industrial monitoring process modelling, but it suffers from trapping in a local minimum during the convergence phase.

Alternatively, to or in combination with ANFIS based models, Empirical Model Decomposition (EMD) tries to decompose the objective signal in Intrinsic Model Functions (IMF) and residuals. Wei [9], proposed a combination of ANFIS based model and EMD. For Wei, the combination of ANFIS and MDE improves the performance of time series analysis with high variability. However, the IMF are decomposed and modelled with one individual model for each signal and it represents a high computational cost, moreover, it can lead to an intense overfitting.

Daniel Zurite *et al.* [10] proposed a methodology based on a multimodal approach, in which the outcome estimation is obtained by the combination of multiple model outcomes that manage different signals dynamics while preserving generalization outputs. They use EMD fo signal decomposition of the target signal and an adaptive dynamic packaging procedure in order to define the number of models to be used. They also use a non-linear mapping procedure to of the available auxiliary signals in order to reduce the dimensionality of the ANFIS convergence problem. Finally, the combination of multiple model outputs is the final forecast.

2.4 Wastewater treatment

With the introduction of IIoT in the wastewater treatment plants, the monitoring of the correct performance of the industrial process have changed deeply in the last few years. Quality control techniques of water monitoring analysis are primarily focused on analytical laboratory tests requiring toxic chemicals, trained personnel and longtime, moreover they cannot be used to indicate changes in plant parameters as they are done after the process is finished. The introduction of the analysis and monitoring of key metrics in wastewater treatment field thanks to the use of data provided by IIoT has been a great advantage in the operability, because it provides a fast, efficient, eco-friendly manner of real-time analysis. Also, the ability of predicting the future state of biochemical parameters, give operators the possibility of changing the parameters proactively to increase the probabilities of a successful process.

Maneesha V Ramesh *et al.* [12] covered the integration of IoT devices in the WWTP field by proposing an IoT based system that can deliver information about the contamination level and the water quality. They introduced sensors to detect hydrocarbons, chemical and metal content, together with pH, conductivity, dissolver oxygen and turbidity to monitor the water quality.

Maged M. Hamed *et al.* [13] proposed a method based in Artificial Neural Networks for predicting biochemical oxygen demand (BOD) and suspended solids (SS) with data obtained in a WWTP in Egypt. They used an exploratory data analysis (EDA) to detect relationships between data and evaluate data dependency. They presented two ANN models, one for predicting BOD and another one for predicting SS concentrations in the plant.

Gokhan Civelekoglu *et al.* [11] proposed six different ANFIS model to assess if those models were a valid input-output model to predict the treatment performance of aerobic biological treatment stage of a full-scale industrial WWTP treating sugar industry wastewater. They implemented three with PCA and three without PCA to estimate the COD, TOC and TN, representing carbon and nitrogen removal.

3 Theoretical Methodology

Even though there are a lot of studies in Industrial Internet of Thing and applied Artificial Intelligence, only few studies have explored the convergence of these two analyses and literature regarding a dedicated methodology for IIoT key industrial metrics prediction is scarce. To go one step further, this project proposes a methodology that tackles the structure, plan and control of the development of a system able to extract meaningful information and predictions of data collected from Industrial Internet of Thing devices using data-driven artificial intelligence model for industrial process monitoring.

Although Data Science applied to Industrial Internet of Things face new challenges related with how IoT extracts data, it can be considered as subset within the data science field and, because of that reason, an applied methodology needs to be used to solve those specific challenges and objectives.

3.1 CRISP-DM Methodology

There are several theoretical methodologies for data science but the two main used are CRISP-DM (Cross Industry Standard Process for Data Mining) and SEMMA (Sample, Explore, Modify, Model, and Assess). Both specify the tasks to be carried out in each phase described by the process, assigning specific tasks and defining what is desirable to obtain after each phase.

Azevedo and Santos [14] compare both implementations and conclude that, even though a parallelism could be stablished between them, CRISP-DM is more complete because it takes into account the applications of the results in the business environment, that is why it has been widely used (during several polls in 2002, 2004, 2007 and 2014, CRIPS-DM was 4 times more applied than SEMMA).

CRISP-DM methodology consists in six phases: (1) business understanding, (2) data understanding, (3) data preparation, (4) modelling, (5) evaluation and (6) deployment. The flow sequence is not rigid and the ability of going forward or backward between phases is critical to make this methodology flexible. Each phase output determines which next phase should be performed and the arrows in picture CRISP-DM methodology flow indicate the most common flow. The external cycle indicated the iterative process of data science projects. It does not end until the deployment has been done, but new iterations can appear after a solution has been found. The phases are described in more detail below:

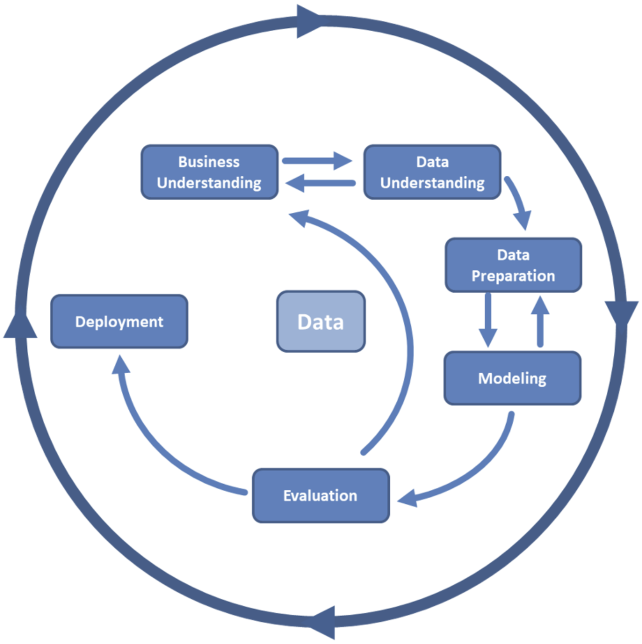
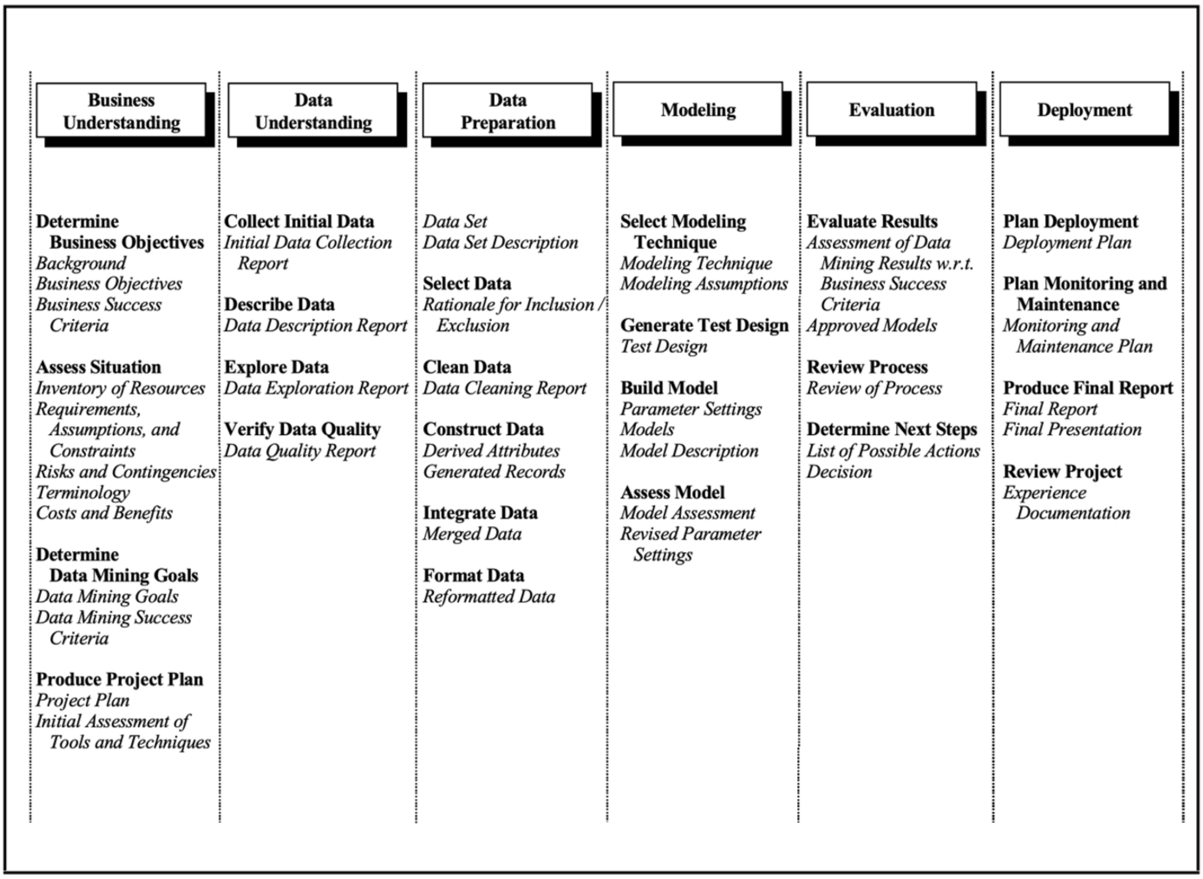


Figure 2: CRISP-DM methodology flow

- Business understanding, requirements definitions: This is the initial phase of the methodology and it focuses on the comprehension of the business logics and the project objectives. Those requirements are converted into a data mining problem and into a plan to achieve the objectives established.
- Data understanding, study of the data: This phase starts with the collection of the initial data, identification of problems related with it and the obtainment of preliminary knowledge to start formulating hypothesis about the hide information.
- Data preparation, exploratory data analysis and feature selection: This phase stablish all the necessary steps to obtain the final dataset, the data used in the modelling phase. This includes tasks related with data cleaning, data transformation, deep data understating and feature selection.
- Modelling: In this phase, statistical and machine learning techniques are applied to the data obtained in the last phase, and parameters selection is performed. Because of some machine learning models need data to satisfy some specific constraints, it is very common to go back to the data preparation phase.
- Evaluation: In this phase, we already have some trained machine learning models that are able to obtain good enough solutions for the problem. Before deployment, it is very important to asses all the workflow until the obtention of the solution and to check that the results obtained are aligned with the initial objectives of the project. After the end of this phase, a decision on the application of the results of the data analysis process should be obtained.
- Deployment: The obtention of a final method that satisfies the initial requirements is not the end of the project, the obtained insights have to be organized and processed

to be presented to the final user. Depending on the initial objectives, the final output could be as simple as a report or a web application.



Business Understanding	Data Understanding	Data Preparation	Modeling	Evaluation	Deployment
Determine Business Objectives *Background* *Business Objectives* *Business Success Criteria* **Assess Situation** *Inventory of Resources* *Requirements, Assumptions, and Constraints* *Risks and Contingencies* *Terminology* *Costs and Benefits* **Determine Data Mining Goals** *Data Mining Goals* *Data Mining Success Criteria* **Produce Project Plan** *Project Plan* *Initial Assessment of Tools and Techniques*	**Collect Initial Data** *Initial Data Collection Report* **Describe Data** *Data Description Report* **Explore Data** *Data Exploration Report* **Verify Data Quality** *Data Quality Report*	*Data Set* *Data Set Description* **Select Data** *Rationale for Inclusion / Exclusion* **Clean Data** *Data Cleaning Report* **Construct Data** *Derived Attributes* *Generated Records* **Integrate Data** *Merged Data* **Format Data** *Reformatted Data*	**Select Modeling Technique** *Modeling Technique* *Modeling Assumptions* **Generate Test Design** *Test Design* **Build Model** *Parameter Settings* *Models* *Model Description* **Assess Model** *Model Assessment* *Revised Parameter Settings*	**Evaluate Results** *Assessment of Data Mining Results w.r.t. Business Success Criteria* *Approved Models* **Review Process** *Review of Process* **Determine Next Steps** *List of Possible Actions* *Decision*	**Plan Deployment** *Deployment Plan* **Plan Monitoring and Maintenance** *Monitoring and Maintenance Plan* **Produce Final Report** *Final Report* *Final Presentation* **Review Project** *Experience Documentation*

Figure 3: Tasks of each CRISP-DM methodology phase

3.2 IIoT CRISP-DM-based methodology

As mentioned before, although this methodology can be applied for IoT analytics systems, the IIoT analytics segments face very particular problems and a more personalized CRISP-DM-based methodology is proposed in order to achieve the possibility of stablish a connection between IIoT and chaotic industrial signal prediction and monitoring. Methodology is divided in 5 different phases, related with CRISP-DM methodology by varying the importance given to some factors: (1) problem definition, (2) IoT infrastructure & data consolidation, (3) data preparation, signal and auxiliar signal analysis, (4) modelling & evaluation and (5) visualization and deployment. This new methodology gives more importance to the IoT infrastructure, as it will define which data is available for analysis and prediction. Also, as this new methodology is focused on achieving valuable insights of chaotic industrial metrics, so a new phase appears whose objective is to find the relationships between objective signals and auxiliar signals in an isolate phase. A more detailed description of each phase of the methodology is given.

3.2.1 Problem definition

This phase focuses on the understating of the problem in terms of IoT. It is important to define the objectives of IoT analytics business case and a baseline understanding which will give a first measurement of the initial situation of the problem provided by the Key Performance Indicators (KPI's). This stablished business case will provide the units to measure the impact of the IoT project in the KPI's, project timeline and project management methodology. The baseline and the measurement of the impact of the project is essential for understanding if the project has reached its initial goals. The definition of the problem and the measurement of the advance is a key part in this initial phase and it should be done with all the stakeholders involved in the project. Sometimes, is one of the hardest tasks to tackle.

Regarding IIoT, a recompilation of all the available IoT devices and data gathered by them have to be created in order to avoid future data missing

3.2.2 IIoT infrastructure and data consolidation

The second phase of the methodology is related with the IoT infrastructure, data availability, data reliability and data collection. In IoT context, data sources are more diverse than in other data science projects and the manner of collecting data can differ between each of the IoT devices. There are two main tasks to be done in this phase: (1) design and select the IoT infrastructure and (2) consolidation of data gathered by the IoT infrastructure. Those tasks have to be performed iteratively until data able to satisfy the necessities stablished in the KPI's are consolidated.

3.2.2.1 IoT Infrastructure

In order to define an IoT architecture, scalability, interoperability, data storage reliability and quality of service (QoS) have to be taken into consideration. There has been a lot of researches in this area trying to attempt to define a universal architecture for IoT. The design of a IoT architecture is driven by six factors:

- Scalability
- Usability
- Connectivity management
- Connectivity management
- Data aggregation and management
- Integrations

Although there are discrepancies, the most used IoT architecture is divided into three layers: (1) perception layer, (2) network layer and (3) application layer [4]. The main idea of this architecture is that the perception layer is the bottom layer of the architecture and it extracts the environment information and transforms it into digital signals, networks layer is responsible of transport the information from the IoT devices through the network and the application layer transfers those digital signals to different contexts.

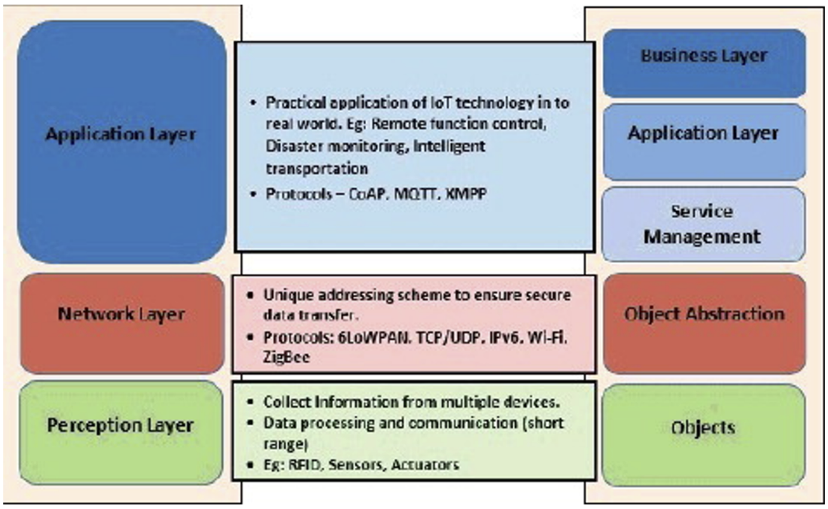

Application Layer
• Practical application of IoT technology in to real world. Eg: Remote function control, Disaster monitoring, Intelligent transportation
• Protocols – CoAP, MQTT, XMPP
Business Layer
Application Layer
Service Management
Network Layer
• Unique addressing scheme to ensure secure data transfer.
• Protocols: 6LoWPAN, TCP/UDP, IPv6, Wi-Fi, ZigBee
Object Abstraction
Perception Layer
• Collect Information from multiple devices.
• Data processing and communication (short range)
• Eg: RFID, Sensors, Actuators
Objects

Figure 4: Layers of IoT architecture [4]

- Perception Layer: This is the initial layer of the IoT architecture, the perception layer is responsible of measure the environment variables that are objective of study, for instance humidity, temperature, pH, concentration of suspended solids… Wireless Sensors Networks (WSN) collects information and process it to get valid data. In this layer there is also a wireless or wired networks that connects the IoT devices using ZigBee, Wi-Fi and many other shot-range communication protocols. As IoT network can become huge, it is very important to identify each sensor with and address that facilitates the communication within the network. RFID, NFC, Bluetooth and 6LoWPAN are used as identification technologies and they facilitate the integration of devices in the network without any extra hassel.

- Network Layer: This is the core part of the architecture; it is responsible of create a secure data transmission environment to send data from the perception layer to the application layer. It has the ability to send information from the sources to different applications and servers. Due to the fact that this layer is the convergence between communication networks and Internet, this layer is the most developed layer in the IoT architecture, combining different communication protocols and technologies. Data filtering and data first processing take place in this layer. As every device has a unique address, the routing ability ease the integration of numerous devices giving the environment the IoT notion. In this layer, medium-large range communication is the standard, so technologies such as Wi-FI, Bluetooth, xDSL, and PLC have contributed a lot to this phenomen. Also, 6LoPWAN have been used to manage IPv6 traffic.

- Application Layer: This is the top layer of the architecture, which serve as a bridge between application and users. This layer is the combination of IoT power with industry expertise to reach and intelligent and sensitive system. For instance, the prediction of plant performance based on industrial process parameters, health monitoring using devices used by the users daily or applications that avoid user capacity on crowded scenarios using cameras. The application layer manages the IoT applications by using standards such us CoAP that runs over UDP protocol, HTML5 websocket.

As mentioned before, the 3-layer architecture is not the only solution to this problem, there are propositions of 4-layer architectures, 5-layer architecture and 6-layer architecture.

The 6-layer architecture introduce three more layers: (1) coding layer, (2) middleware layer and (3) business layer. Because of the importance given to the unique identification of IoT devices, coding layer appear as the layer which manages that identification. The processing of the information gathered by the sensor is processed in the middleware layer, that uses technologies as Cloud Computing and Ubiquous computing to ensure the access to the database. The top layer becomes the business layer that is in charge of managing the whole IoT application environment.

IoT infrastructure provides the systems with the information needed for the development of the intelligent system. Given the problem definition and requisites specification obtained in the Problem definition phase, the IoT architecture stablish which IoT devices are collecting environment measurements that are important for the system goal achievement. For this reason, IoT architecture and data consolidation are very related and are considered in the same phase. Iteration over the design of the IoT architecture and data consolidation validation is common and should be done in order to assess that all the information needed for the successful deployment of the system is achieved.

3.2.2.2 Data consolidation

Data consolidation tries to extract the information gathered by the IIoT devices and to consolidate it in the storage system. There are two types of data processing systems that will help running two different system functionalities, while On-Line Transactional Processing (OLTP) systems are purely operational, On-Line Analytical Processing (OLAP) systems uses data to gain valuable insights (not predictions).

Going deeply into OLTP systems, they enable real-time execution of large database transactions, as mentioned before, they are purely transactional. They usually run over a transactional database that is in charge of store data and its changes. They usually support all the data transactions and feed periodically the OLAP systems via extraction, transformation and loading (ETL) processes. Their purposes are:

- They provide support for low-cost transactions such us insertions, deletions and updates.
- They can offer multi-user access without risking data integrity.
- They offer very rapid processing.
- They have methods of providing rapid searching, retrieval and querying.
- They need to be available all the time, backups are critic.

Otherwise, OLAP systems provide multi-dimensional analysis with high-speed at large volumes of data, OLAP systems perform their operations over a data warehouse, data mart or other data store. There are excellent systems for analytical purposes, data mining, business intelligence and reporting.

OLAP databases are based on OLAP cubes, which allows rapid querying, reporting and data analysis. Those OLAP cubes are made of data dimensions, usually business parameters, that acts like an index identifying values within a multidimensional array and provide a very concise manner of organizing data for exploration and retrieval. The OLAP cube also extends the table format of a OLTP database, different operations such as drill-down, roll up, dice, slice and pivot are used to traverse over the cube to gain valuable information.

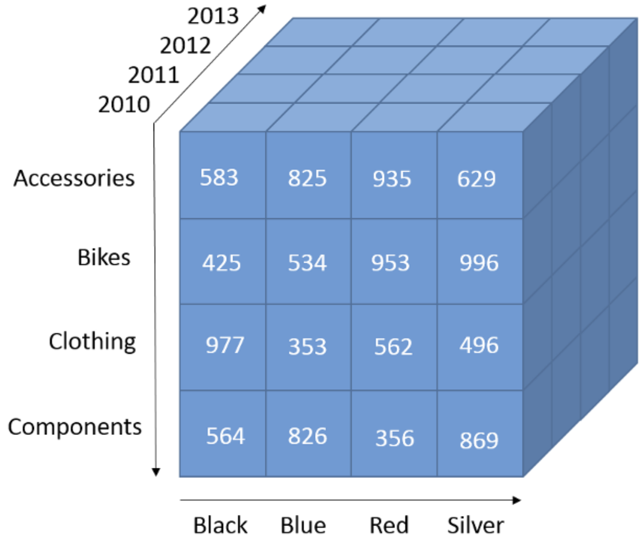


Figure 5: OLAP cube

3.2.2.3 IoT Infrastructure and Data Consolidation Diagram

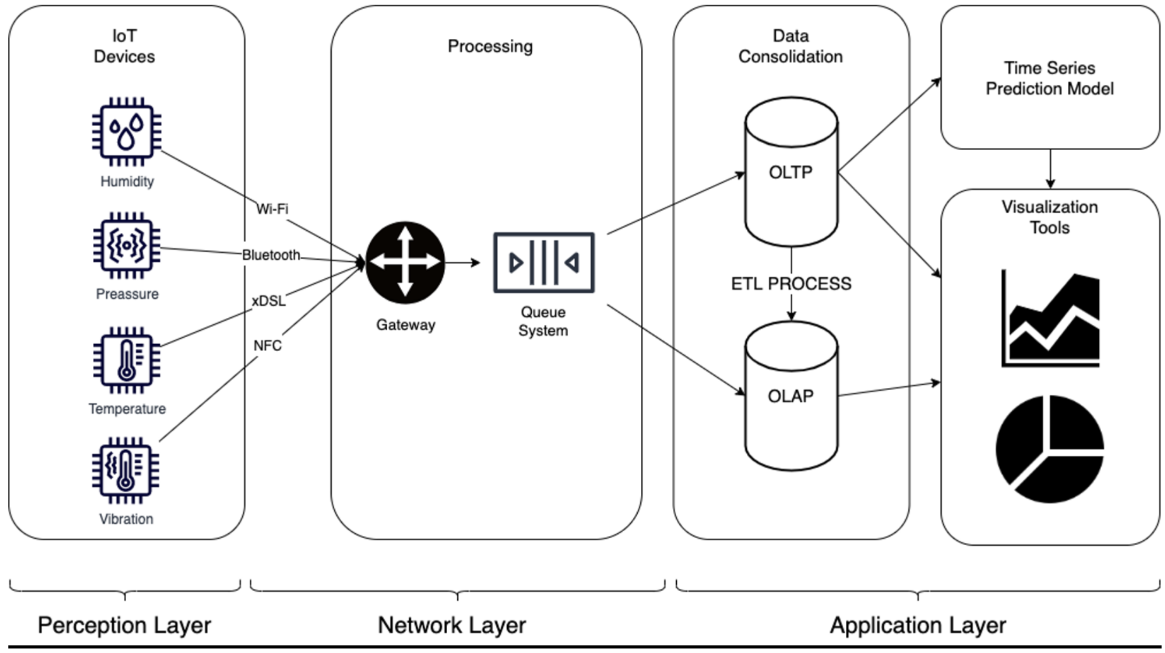


Figure 6: IoT Infrastructure

3.2.3 Data preparation, signal and auxiliar signal analysis

This data preparation phase is very similar to the data preparation phase of CRISP-DM methodology. Exploratory data analysis (EDA) should be performed, together with feature selection. However, with the IIoT introduction, the convergence between data preparation and IIoT infrastructure is higher and the iteration between these two phases is high.

Although data understanding is commonly done before the data preparation phase, in this particular case, with the introduction of the chaotic industrial metrics monitoring and prediction objective, this phase is vital for the satisfaction of the goals. In this phase, it takes place the analysis of the industrial environment regarding the parameters that model the industrial process.

In industrial environments combined with IoT, the number of signals and industrial metrics gathered is large and, therefore, the establishment of a multivariate analysis is crucial to achieve good results. Signal and auxiliar signal analysis can be defined as a multivariate time-series analysis problem in which multiple time-series data (signals and auxiliar signals) contribute to the output of the objective signal. Not only does the past records of the signal itself modulates the outcome, but the nature of other auxiliar signals influences.

3.2.3.1 Signal analysis and chaotical time-series analysis

Once we have defined the problem as a multivariate time-series, another critical aspect is the sampling frequency of the time entries. Sampling frequency is one important aspect to take into account in IIoT infrastructure and data consolidation because it can lead to inconsistencies of time stamp misleading between data and this is a problem that need to be avoided in previous steps. Sometimes, the nature of the signal and the industrial environment will tell which frequency does it need to be sampled, but a homogenization has to be performed in data preparation step if time-series frequency misleading is found.

Another important aspect regarding time-series signals is if it is stationary or non-stationary. Stationary time-series data is the one whose properties does not depend on the time at which the data is observed, therefore, data with trends or cycles are non-stationary. When performing statistical time-series modelling, stationary data is more likely to show better results than non-stationary data due to the fact that summary statistic do not change over time and assume they will not change at the prediction series values. Seasonality can be another factor that make time-series data non-stationary, it means that there are alterations in the signal values that occur at a specific regular time intervals. In order to check seasonality, autocorrelation is used to check the correlation with the data with a lagged version of it as a function. The autocorrelation function (ACF) calculates values for a range of lags, and the correlation amount those lags defines the seasonality. Also, non-linear autocorrelation can be checked with other tests such us Durbin-Watson test. Industrial metrics signals seem to be chaotic and to present non-linear relationships with other signals, but it depends on the nature of the industrial process.

Although the IoT infrastructure has to be reliable and avoid the bad operation of the IoT sensors, there is the possibility of having missing data because a temporal malfunction of the system. In order to solve missing data related problems, a time subset can be chosen but it could result in losing the intrinsic nature of the signal and to not be able to model it correctly. The are some methods to interpolate the values of the time-series such us mean interpolation, median interpolation, mode interpolation, linear interpolation or Spline interpolation, Last Observation Carried Forward (LOCF) or Next Observation Carried Backward (NOCB).

3.2.3.2 Auxiliar Signal Analysis

As mentioned before, multivariate time-series analysis lies on the possible correlation between two or more signals, that is why another multivariate correlation methods should be perform to check causality within variables.

The cross-correlation is the correlation between main signal and lagged versions of auxiliar variables, but it is not very suitable for testing causality because it can lead to problems related with autocorrelation as it leads to peaks in the cross-relation, even if they are truly independent. Because of that, another more suitable method such us Granger causality is used instead. The idea behind Granger causality is to predict the value of the objective signal by using two models, the first using the objective signal itself, and the second uses the past values of the objective signal and the auxiliar signal. If using the past values of the auxiliar signal improves the results of the prediction beyond using the objective signal alone, a Granger causality is found. There are also multivariate versions of Granger causality that allows to check causality between more variables. Although the Granger causality is the most used method, there are other like sim causality, intervention causality and structural causality.

Also, there are other methods like Principal Component Analysis (PCA) that can be used to reduce the dimensionality of the signal dataset but also for displaying some kind of clustering if you look to the data in the visual principal components. There are some rules in order to apply PCA to time-series data. Firstly, each time-series variable has to represent a single feature, this is, the objects represent data throughout the time interval and the feature itself represents the time-series. So, each time series represents an entire feature whose values are spread out over the rows.

3.2.3.3 Data preparation

Apart from all the corrections methods applied in a normal EDA process, time-series data need to be treated particularly regarding processing because time-series prediction models need the data to fulfil particularly assumptions in order to achieve a good result. The most common steps in time-series data preparation analysis are:

- Make data stationary: The objective of this step is to remove both trend or seasonality within the data. There are several approaches to make data stationary:
  - Differentiate the series values: The differentiation means that for a time-series X, the value differentiation generates a new series z where:

    $$z_k = x_t - x_{t-1} \tag{1}$$

    There could be different steps within this method regarding differentiation of lineal trend first, quadratic trend then and so for.
  - Difference the time entries: Time entries can be differentiated from a certain occasion and receive a new series that is scaled to this event. This could lead to another metrics, being the case of study, if the amount of carbon is high and some alerts are shown, we can be interested in scaling the time alert to the detection time in order to check the reaction time.
  - Remove trend: To remove trend the fitting of a linear regression to model the trend and to continuing modelling the residuals that are cleaned from the trend. But there is a big problem, not all signals can be modelled by fitting a linear regression model, moreover, it is almost impossible if they are chaotic or based of a multiple number of factors such as industrial signals.
  - Remove seasonality: If there was detected a seasonal lag, difference series values method can be applied by obtaining a new time-series z where:

    $$z_k = x_t - x_{t-seasonality\_lag} \tag{2}$$

    It can be also used the seasonality pattern detected to calculate the average pattern values and create a new series such us:

    $$z_k = x_t - x_{avg\_values} \tag{3}$$

  - Drive a logarithmic or exponential transformation: Those transformations will help to stabilize the variance of the series across the time.

  After any of these methods have been applied, autocorrelation test has to be done in order to check the performance of the transformations.

- Treat missing values: There are some time-series prediction methods that can handle missing values by theirselves but it is not the case for others. As mentioned before, after detecting missing data in the time-series a different number of methods can be applied to try to solve it:

  - Impute the entries: Interpolate the data using a linear regression to each time-series variable to complete the missing entries by calculating the curve. Another option is to use spline interpolation, that fits more than one curve to obtain the missing samples or to use a machine learning model able to predict those missing values.

  - Remove samples or features with missing information: It is not useful to maintain variables that could be recorded throughout the time. But it is more difficult to remove a row sample because it does not have enough data, because you do not know how much data is enough.

  - Select a period of time: There are periods of time that could be without data because different reasons such us maintenance, system or sensor failure. Select a period of time would be latest option if we have to get rid of missing values because by selecting a subset, we can be broken the intrinsic nature of the time series.

  - Let the model handle it if it is able to.

- Multi-frequency data to uni-frequency data: While working with different industrial IoT sensors, there are variables that might have different sample frequency than others. For example, temperature of the water tank could have its frequency in hours meanwhile the pH could be working in seconds or minutes. There are some models that assume that time-frequency is all the same for the variables and do not accept variations between variables frequency. In order to avoid this problem, we have to choose between working in low frequency or high frequency.

  - Work with low frequency data: Summarize over the highest frequency to achieve a new low frequency series for all the signals. For example, calculate the main average temperature for the water tank in a day to reduce the frequency. This could lead to the loose of important information, depending on the objective to study.

  - Work with high frequency: Expand the low frequency, for example expand the water tank temperature to minutes or seconds, although it will produce redundant data, we will not lose any information regarding the signal.

After finalizing with this phase, we will have a complete understating of how much an objective signal is autocorrelated and which other signal influence the value of that objective signal. Moreover, a cleaned and pre-processed data will be ready for fitting a model and the evaluation phase.

3.2.4 Modelling and evaluation

A great number of industrial signals are chaotic, which means that is not periodic (but could be stationary), has random time evolution and broadband spectrum and it is produced by a deterministic non-linear dynamical system with an irregular behaviour. Chaotic time-series prediction techniques address the problem of forecasting horizons of chaotic data. This kind of predictors are focused on compacting the description data and guarantee a low error regarding mean squared error (MSE) and root MSE (RMSE).

They are several techniques that try to solve the problem of chaotic time-series analysis considering both short-term and long-term prediction horizons. Those methods are artificial neural networks (ANNs) [15] [16] [17], XGBOOST Regressor [28] [22], support vector machines (SVMs) [24] [25] and hybrid systems [10] [26].

3.2.4.1 Aritificial Neural Networks (ANN's)

ANNs are mathematical representations that tries to imitate the human brain and its way of processing the information. Usually, ANNs are composed by elementary computing units called neurons and their processing capacity is stored at neural connections. The simplest structure of an ANN is formed by an input layer, one or more hidden layers and the output layer. Each hidden layer is composed by one or more neurons that provide the objective value. The structural form of a neuron is shown in Structure of a neuron.

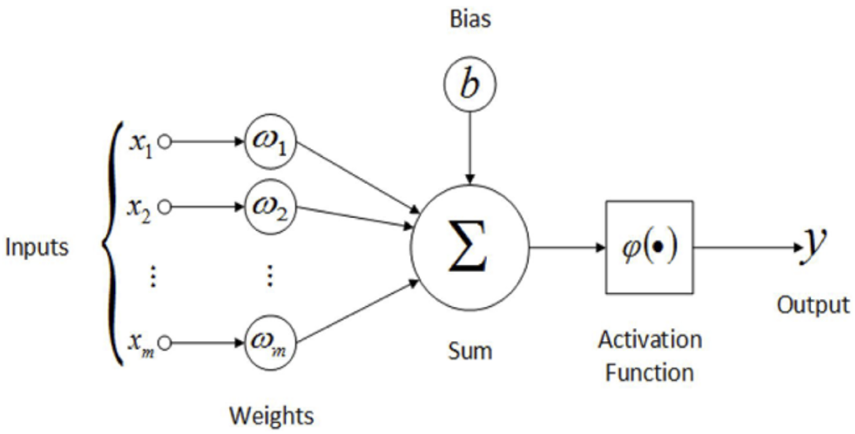


Figure 7: Structure of a neuron

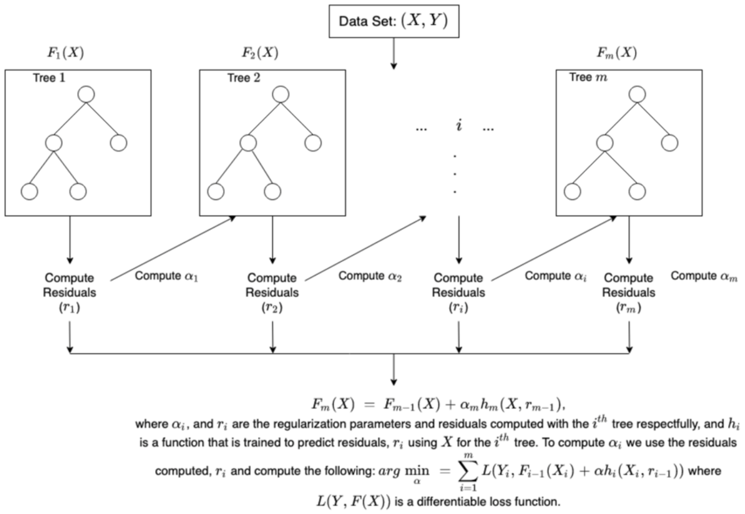
Takens [33] stated that there is a function with at most 2d + 1 past measurements of time series that allows the forecasting of future values and that prediction would be as good as solving the system with all the degree of freedom. Although Takens did not set the function that give the desire future extrapolation, an ANN can be successfully used by setting a future temporal delay of x(t + delta) and the input temporal time series. ANN can be trained with a certain number of iterations [32]. In order to choose the structure of the ANN corresponding to the number of neurons and layers of it, there is the geometry pyramid rule that stablish that the hidden layers have to have less neurons that the input layer. It seems to be a consensus in which activation function the neurons depending on the layer, being the linear function the one chosen for the output layer and the hyperbolic tangent function for the rest of layers [32] [31].

3.2.4.2 XGBOOST Regressor

XGBoost is a popular and efficient open-source implementation of the augmented trees algorithm. Gradient boosting is a supervised learning algorithm that attempts to adequately predict a target variable by combining estimates from a set of simpler and weaker models.

When gradient boosting is used for regression, the weak learners are the regression trees, and each regression tree assigns an input data point on one of its leaves that contains a continuous score. XGBoost minimizes a regularized objective function (L1 and L2) that combines a convex loss function (based on the difference between the objective and the predicted outputs) and a penalty term for model complexity (i.e., the tree functions of regression). The training proceeds iteratively, adding new trees that predict the residual errors of the previous trees, which are then combined with the previous trees to make the final prediction. It is called gradient boosting because it uses a gradient descent algorithm to minimize loss when new models are added.

Below is a brief illustration of how the gradient tree boost works:



$$F_m(X) = F_{m-1}(X) + \alpha_m h_m(X, r_{m-1}),$$

where α_i, and r_i are the regularization parameters and residuals computed with the i^{th} tree respectfully, and h_i is a function that is trained to predict residuals, r_i using X for the i^{th} tree. To compute α_i we use the residuals computed, r_i and compute the following: $arg \min_{\alpha} = \sum_{i=1}^{m} L(Y_i, F_{i-1}(X_i) + \alpha h_i(X_i, r_{i-1}))$ where $L(Y, F(X))$ is a differentiable loss function.

Figure 8: XGBOOST Regressor schema.

3.2.4.3 Support Vector Regressor

The main characteristics of Support Vector Machines is that it is a Kernel Method so it is uses kernel functions that describe a problem in a characteristic high-dimensional space and applies linear operations to nonlinear problems. A data mapping from the input space to a. high dimensional space is done by a kernel function that computes the dot product in the feature space. Regarding time-series problems, the Kernel function $k(x, y) = x \cdot y$ has been very useful and the decision function is of type $f(x) = wx + b$.

Usually, the used SVM is least squares support vector machines that is a new technique for regression problems (LS-SVM). LS-SVM is trained using time-series data as inputs as a single value for the output representing the value of the system in a given time instant. The method of LS-SVM for time-series prediction is also known as sliding window. The choice for the kernel function can vary but RBF kernels usually perform well.

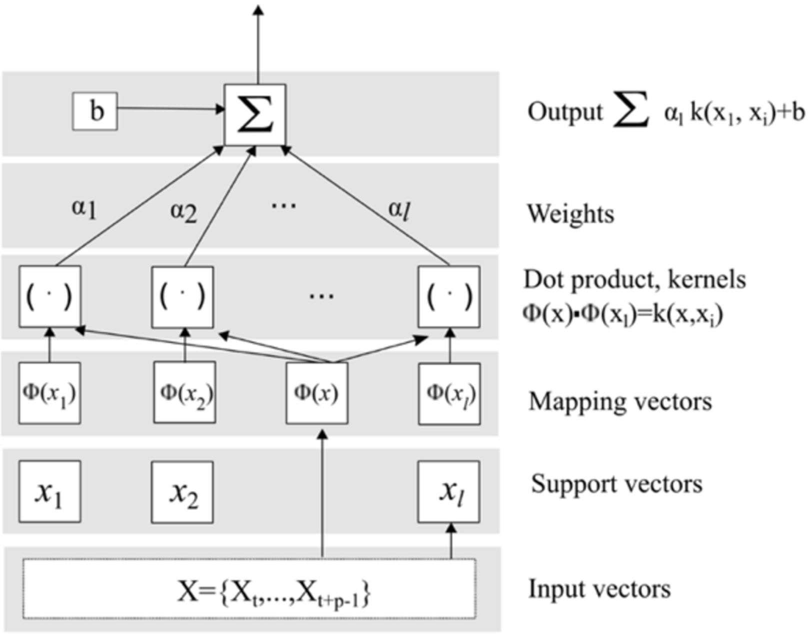


Figure 9: SVR structure

3.2.5 Visualization and deployment

3.2.5.1 Visualization of time-series data and time-series predictions

Time-series data visualization methods mainly focus on how values of data attributes vary over the time. There are two type of time-series data visualization methods: (1) static display and (2) animation methods. Static display refers to the combination of multi-views to show the evolution of data over the time, showing trends and the intrinsic law within the time data. Animation methods records the state of the data in a slice of time and the show data values in function of the chronological slice.

Both methods have advantages and disadvantages based on the goal we want to achieve, static displays are more intuitive and effective, moreover, it avoids the over additional cognitive load to the user. Animation methods are more align with the human being vision, being able to distinguish different time-frames, but it may take a lot of time between animations and it can lead to a blurred view of the data trend. Usually, a combination of both methods is used in order to achieve effectiveness and analytical granularity.

In this methodology, we are going to mainly focus on the two methods reviewed by Yujie Fang *et al.* [30]: (1) visualization of time attributes and (2) visualization of high-dimensional time-series data. Both methods depend on the characteristics of time-series data to be analysed and its attributes, Yujie Fang mention three manners of differentiation time-series attributes:

- Linear time and cycle time: Represents the differentiation between data that is stational of has trends that are repeated in cycles and linear data that change linearly over time.
- Time points and time intervals: Time points corresponding to discrete points and time intervals being small linear time domain.
- Sequential time, branch time and multi-angle time: Sequential time are values that are given in chronological order, branch time are values that can refer to main lines in the time-frame and multi-angle time are values that describe different views of the time.
- Time attribute visualization methods: As mentioned before, visualization of time-series data is different depending of the data attributes, but the standard visualization consists in the x axis representing the time-frame and y axis representing other variable values, this is known as line chart. With these plots, sometimes it is difficult to express the periodicity of time so the are other methods such us the spiral diagram method, calendar view, theme river view and dynamic visualization.
  - Spiral diagram: This diagram is very optimal for analysing periodic data. It is a spiral, as its name suggests, and each circle represents a cycle. Regarding chaotic time-series data, it is not the most suitable method because one assumption made over chaotic data is to not be non-stationary, that means not to have cycles.
  - Calendar view: This view allows to divide the time into time-frames according to the wanted granularity such us hour, day, week.
  - Themeriver view: This method uses the river flow shape to represent the time progress given by the width, direction and colour of the river. Themes are stacked over a spiral curve the its function is to represent the time axis.
  - Dynamic visualization: This method can show the data trend by showing frozen data frame by frame.
- High-dimensional time series data visualization methods: These methods have to show data state changes over the time and the changes done by each dimension with time. Usually, the most used time-series visualization methods include a rivertheme and a parallel coordinate chart with a time axis. This leads to a chart in which different attribute values changing over the time are visualized together.

3.2.5.2 Time series visualization tools

Time series plotting tools usually have preconfigured dashboards. The most popular time-series visualizations tools such as InfluxDB and Grafana are very suitable for visualizing time-series data and for providing different ways of presenting meaningful insights on time data. Usually both are used in combination because their integration allows to visualize data from different sources.

User Interface from InfluxDB includes a user-friendly mode to deal with write data into InfluxDB, visual scripting and querying tools, moreover, it includes the possibility to perform data transformation tasks and alert creation tools. The visualization horizon of InfluxDB includes different types of charts and plots such us gauge charts, line charts, histograms, heatmaps, scatter plots and table, in addition, it also includes also different types of powerful types:

- Visualization of time-series data using custom graphs from plotting libraries such us Plotly and Dygraphs.
- The utilization of templates that includes a packager and set of pre-loaded dashboards.

Grafana allows the integration of different data sources in a straight-forward manner, also includes data source plug-in for InfluxDB. It extends with rich graphing features the plots available only with InfluxDB by offering high level of customization for dashboarding building and editing. It gives:

- Dynamic and reusable dashboards.
- Data exploration and dynamic drill-down.
- Logs exploration.
- Visually defining alert rules.
- Annotations to view event metadata and tags.

3.2.5.3 Deployment

After achievement a model that satisfy the definitions done in the problem definition phase, the systems have to be deployed in a way that it is useful to the organization. This implies the integration of the IoT architecture, models and visualization mechanism in the organization. This task usually is divided in four parts:

- Deployment planification: it is necessary to stablish a plan to introduce the system in the organization information systems in a way that is extends the information and operation available.
- Maintenance planification: Maintenance and monitorization of deployment of the systems is an importance step in the integration within the industrial company business. A good maintenance planification helps to avoid underperformance and the bad use of the analysis results.

- Final report: It is necessary to write the documentation of all the project and it must contain all the deliverables and the result obtained in each phase. This phase usually contains a final presentation or meeting to conclude the project.
- Project revision: This task tries to assess the correct deliver of the project and to understand which were the difficulties of the project in order to achieve that knowledge and to avoid future related problems.

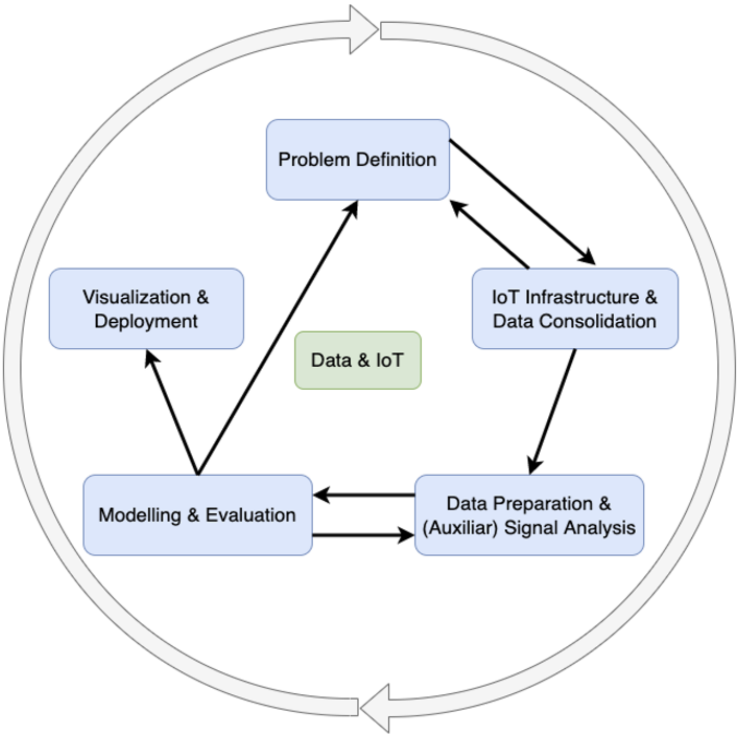

Problem Definition
IoT Infrastructure & Data Consolidation
Data Preparation & (Auxiliar) Signal Analysis
Modelling & Evaluation
Visualization & Deployment
Data & IoT

Figure 10: Methodology for IIoT + Data Science diagram

4 Practical use of the methodology

In this section, a practical application in a real problem is going to be addressed with the know-how established in the theoretical methodology. As mentioned before, the field of the problem is going to be wastewater treatment process which is a potential candidate to be affected by the premises and solution established in the last section. In addition, the installation of a network of IIoT devices, gives us the backbone on which to develop the practical case with more chances of obtaining a successful result.

As mentioned, we are going to develop this practical case under the guidelines proposed in Theoretical Methodology. This means to develop each one of the phases proposed with their inputs and desirable outcomes.

4.1 Problem definition

Water treatment processes have become a key facility in minimizing the ecological impact that industrial and human activities have on the planet's water reserves. The use of digital technologies such as IOT and artificial intelligence allow these types of facilities to take advantage of the data generated to be more efficient and improve the quality of the water they release into the sea.

One of the most critical points in water treatment plants is to be able to know in advance how the process is going to behave with respect to the water we currently have in the input tank, since it takes between 24-27h to see the water quality we will obtain at the output, when it is already too late to establish corrective measures. The objective of the methodology developed in this project is to predict the evolution of key metrics in the industrial process in order to act beforehand those metrics are out of range. For then, develop a multivariate chaotic signal prediction model that, based on current operations, is able to forecast the carbon removal performance (CRP) and its value (TOC). Also, a deliverable whose objective is to visualize the industrial metrics together with TOC and CRP, in order to help to understand the causes that make the TOC and CRP deviate from the desired range, being able to identify which cases provide poor performance in this process.

4.1.1 Characterization of the wastewater industrial process

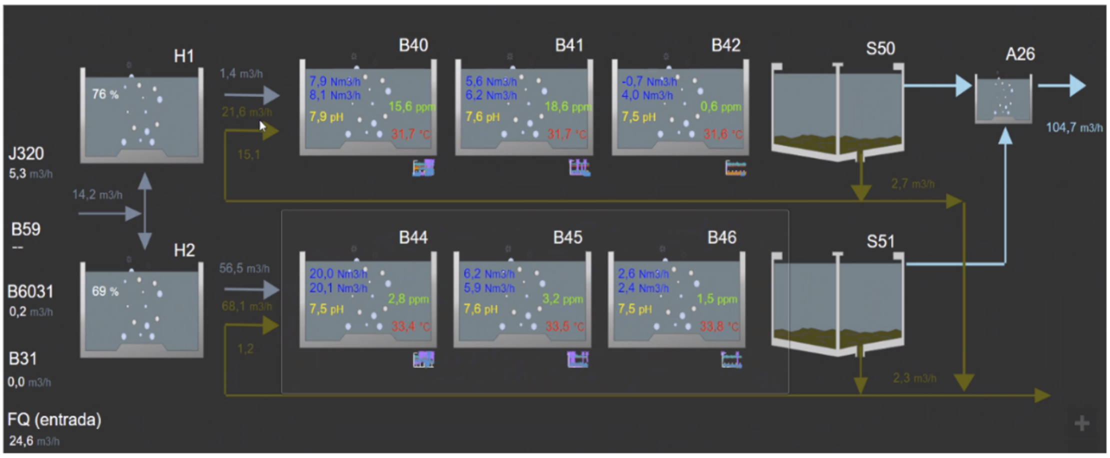


Figure 11: Wastewater treatment industrial process flow.

Industrial wastewater treatment facilities are usually made up of a succession of physical-chemical and biological processes, both aerobic and anaerobic via, that complement each other and allow comprehensive treatment to be carried out under the best technical and possible economics. The objectives of the plant are:

- Elimination of residues, oils, fats, floating matter or sand and evacuation to the appropriate final destination point.
- Elimination of organic and/or inorganic settleable materials.
- Elimination of ammoniacal and phosphorus-containing compounds.
- Transform the retained waste into stable sludge and ensure that it is properly disposed.

Usually, the processes in a WWTP plant are grouped as:

- Water lines: Pre-treatment, Input, Treatment, Output and Post-treatment.
- Sludge line: Thickening, Digestion, Conditioning, Drying and Elimination.
- Gas line: Methane production.

The objective of this practical case addresses the analysis of the water line as the data available for study corresponds to this part of the process. In order to understand better the industrial process and the data available, a schema of the Water Waste Process object of this application and their associated parameters is given:

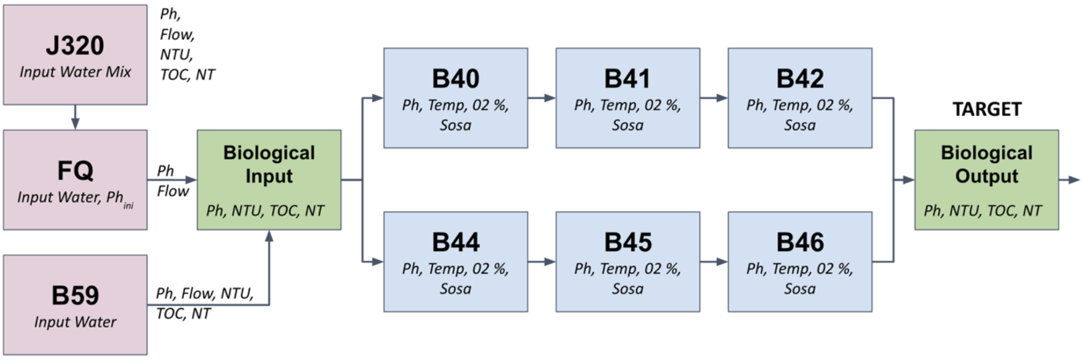


Figure 12: Characterization of the WWT industrial process with ponds and parameters measured.

As we can see in Characterization of the WWT industrial process with ponds and parameters measured figure, we can distinguish three different parts within the industrial process: (1) pre-treatment phase, (2) input phase, (3) treatment phase and (4) output phase. Pre-treatment: it is composed by 3 ponds (J320, FQ and B59) whose objective is to pre-treat the water input in order to maintain water parameters within a range that makes the water treatment process able to remove the organic material. This the phase in which operators can move water between ponds and decide which pre-treatment method apply. Water parameters measured in this phase are directly correlated with the water quality emitted by the plant, so it is essential for the operator stablish how the input parameters will affect the output water to decide which action to take.

Input phase: This phase collects all the water from the pre-treatment ponds (biological input pond) and brings them together to start the organic carbon removal process. In this phase, the measurements of the indicators that allow establishing the performance of the plant as well as the quantity of water and its quality that enters the process.

Treatment phase: This phase is made up of two separate lines. These two lines have 3 rafts each fulfilling a specific function each (B40, B41, B42, B44, B45, B46). Each of these ponds contain detailed information on the concentration of oxygen in the pond, the pH, the temperature, the soda input and the water flow rate. The organic carbon removal process is carried out by cultivating a colony of bacteria responsible for carrying out this task. The chemical and physical parameters of the ponds are directly related to the performance of the bacterial colony in removing carbon from the water.

Output phase: In this phase, parameters that indicate the performance in the elimination of organic carbon carried out in the process are defined. The water obtained in this phase is the result of the process and although post-processing can be carried out, it should comply with the established chemical and physical requirements to avoid additional costs.

It is also important to establish the time sequence of the process that allows us to monitor the water over time so that the measured parameters correspond approximately to the same water sample. Talking with the plant operators, a time lag in the process has been established as shown in the following figure.

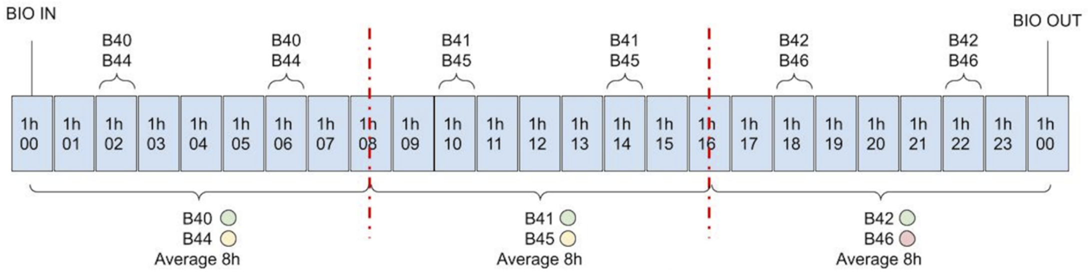


Figure 13: WWTP Time Lag: B40 and B44 have 2 time lags, B41 and B45 have one time lag, accordingly to Biological Output measurements. Process time is 24 hours so it can be seen as 3 parts of 8 hours each, corresponding to each time lag.

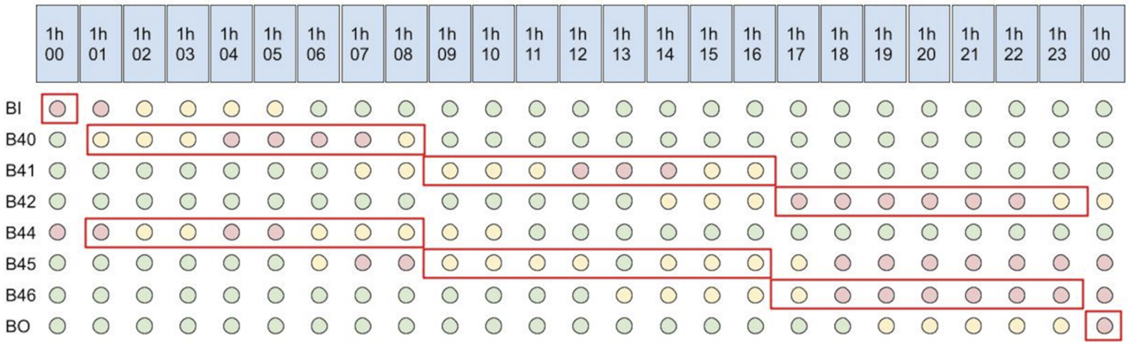


Figure 14: Description of how to follow the same water sample throughout the whole water line process.

As we can see, the time required for a water sample to enter the system, be treated and leave is 24 hours, so the difference between the same sample taken at the biological input and output is this time range. Establishing this offset allows us to create the organic carbon degradation performance measure, which is essential to know if the plant is operating correctly.

4.1.2 Objectives and key metrics

The purpose of the proposal has two distinct objectives:

- To develop a data-driven model able to predict future TOC and CRP values that allow certain reactivity and proactivity to plant operators to make changes in the parameters of the industrial process to correct possible deviations. As mentioned in the characterization of the problem, the prediction horizon should be 24-48 hours to have enough time to pre-treat the water if there is a risk of deviation.
- Develop a visualization application which is able to show the relations between the industrial process mechanism and the predictions made by the data-driven model.

The key metrics of the project are the value of total organic carbon (TOC) in [ppm] and plant carbon removal performance (CRP). Definitions of the key metrics are the following:

- Organic Carbon Degradation (TOC): Value in ppm measuring the presence of organic carbon in the water.

$$TOC_t = Bio\_Out\_TOC_T \tag{4}$$

$$TOC_{t+24h} = Regression\_Model\_TOC_{t+24h} \tag{5}$$

$$TOC_{t+48h} = Regression\_Model\_TOC_{t+48h} \tag{6}$$

- Carbon Removal Performance: measurement of the carbon elimination within the industrial process.

$$CRP_t = \frac{Bio\_Out\_TOC_t}{Bio\_In\_TOC_{t-24h}} \tag{7}$$

$$CRP_{t+24h} = \frac{Bio\_Out\_TOC_{t+24h}}{Bio\_In\_TOC_t} \tag{8}$$

$$CRP_{t+48h} = \frac{Bio\_Out\_TOC_{t+48h}}{Bio\_In\_TOC_{t+24}} \tag{9}$$

These key metrics will be predicted by the developed models, establishing a model for a 24-hour prediction and another model for a 48-hour prediction. These time margins allow sufficient anticipation for the operators to carry out the pre-treatment of the water to avoid poor degradation of the organic carbon.

4.1.3 Risks identified

The main risks identified that may condition the success of the project are the following:

- Data availability: Data of the process need can be gathered and extracted from databases. Process data is the starting point for the posterior analysis and methodologies object of the project. For this reason, it is mandatory for us to be provided with the data from the PLC's and the different equipment of the WWT plant, with important access to data from IIoT devices.
- Data traceability: In order to be able to carry out the project correctly, we must be able to temporarily synchronise the continuously recorded data of the process (PI). If any data cannot be traced, it will put under risk the analysis.
- Data quality and variability: The data collected must be of sufficient quality in terms of completeness of the records, and sufficient variability to represent the state of the water treatment process. We will carry out a quality study of the data obtained in the project to ensure this point (phases: Data Preparation and Data Consolidation).

4.2 IIoT infrastructure and data consolidation

4.2.1 IIoT infrastructure

Unfortunately, it has not been possible to access the IIoT devices that take measurements directly from the plant because of security reasons.

On the contrary, it has been based on a historical extraction of an Osyshoft PI database. For the use of the proposed methodology, a data extraction process (ETL) has been recreated, which is responsible for loading them into a temporary database, imitating the process that would be carried out in a real infrastructure. This architecture has allowed to establish the basis for carrying out the analysis of the industrial process. creating all the logic assets involved in the IIoT infrastructure. Below is a detailed schema of the implementation carried out and each of the parties involved.

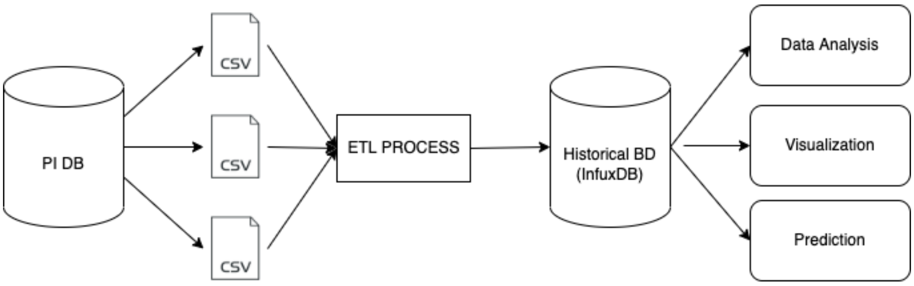
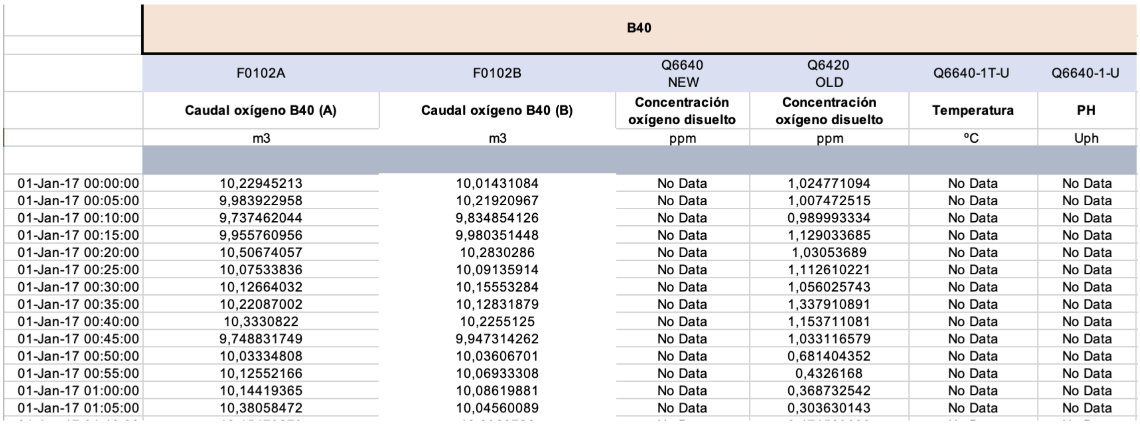


Figure 15: Recreation of the IoT infrastructure to simulate the the real IoT workflow.

4.2.2 Data consolidation

As mentioned above, the data has been made available through files in excel format. Due to the amount of data available and the heterogeneity of the files, it has been decided to implement an ad-hoc extraction, transformation and loading process that allows the necessary operations to be carried out to ensure the highest degree of quality and integrity of the data before inserting them in the database.

The available files contain temporary information of one or more ponds each. In the header of the file, the origin and the measured parameter are indicated. The sampling frequency indicated in all the files is 5 minutes, which is high for the described dynamics present in the industrial process and the prediction objective of 24-48 hours.



	B40					
	F0102A	F0102B	Q6640 NEW	Q6420 OLD	Q6640-1T-U	Q6640-1-U
	Caudal oxígeno B40 (A)	Caudal oxígeno B40 (B)	Concentración oxígeno disuelto	Concentración oxígeno disuelto	Temperatura	PH
	m3	m3	ppm	ppm	ºC	Uph
01-Jan-17 00:00:00	10,22945213	10,01431084	No Data	1,024771094	No Data	No Data
01-Jan-17 00:05:00	9,983922958	10,21920967	No Data	1,007472515	No Data	No Data
01-Jan-17 00:10:00	9,737462044	9,834854126	No Data	0,989993334	No Data	No Data
01-Jan-17 00:15:00	9,955760956	9,980351448	No Data	1,129033685	No Data	No Data
01-Jan-17 00:20:00	10,50674057	10,2830286	No Data	1,03053689	No Data	No Data
01-Jan-17 00:25:00	10,07533836	10,09135914	No Data	1,112610221	No Data	No Data
01-Jan-17 00:30:00	10,12664032	10,15553284	No Data	1,056025743	No Data	No Data
01-Jan-17 00:35:00	10,22087002	10,12831879	No Data	1,337910891	No Data	No Data
01-Jan-17 00:40:00	10,3330822	10,2255125	No Data	1,153711081	No Data	No Data
01-Jan-17 00:45:00	9,748831749	9,947314262	No Data	1,033116579	No Data	No Data
01-Jan-17 00:50:00	10,03334808	10,03606701	No Data	0,681404352	No Data	No Data
01-Jan-17 00:55:00	10,12552166	10,06933308	No Data	0,4326168	No Data	No Data
01-Jan-17 01:00:00	10,14419365	10,08619881	No Data	0,368732542	No Data	No Data
01-Jan-17 01:05:00	10,38058472	10,04560089	No Data	0,303630143	No Data	No Data

Figure 16: Extract from B40 historical file, in which it can be seen the header (B40), parameters (F0102A, F012B, . . .), descriptions, the units (m^3, ppm, . . .) and the values.

Once the ETL process has been done, data was loaded into an InfluxDB database that is ideal for time-series databases (TSDB), which store time series. These databases are used, among other things, to store and analyse sensor data or protocols with timestamps over a certain period of time. For example, Internet of Things devices or scientific measuring instruments deliver millions of incoming data sets in a constant stream of data. Compared to ordinary relational databases, TSDBs like InfluxDB offer clear speed advantages when it comes to storing and processing time-stamped measurement data. A traditional DBMS slows down when organizing complex indexes, which are not used at all in this area of application. InfluxDB can maintain high write speeds over a long period of time because it uses a very simple index.

Once the loading process has been carried out successfully, we find data corresponding to the period between 01-01-2017 and 30-06-2022 which, with a frequency of 5 minutes, correspond to 577,944 samples, which makes the database quite extensive to carry out the objective analysis. In addition, the number of characteristics is also extensive, comprising a total of 103 variables corresponding to the water lines of the treatment process to be analysed. Despite this large number, not all the variables and not all the samples will be used. On one hand, only the interesting variables for the organic carbon degradation process will be used, which correspond to Characterization of the WWT industrial process with ponds and parameters measured figure, and only the longest period will be chosen where the data integrity and quality is the highest possible.

4.3 Data preparation, signal and auxiliary signal analysis

4.3.1 Data preparation

In this phase, the variables that are interesting for the analysis of organic carbon removal and for the operation of the water lines will be chosen, as well as the time window that allows the analysis to be carried out.

The time window to be analysed and modelled is the one corresponding to 2019 due to the fact that it has the two important factors. 2019 has a percentage of low null values that corresponds to 6.11% and also has a percentage of 0.05% of null values in the target variable. The table below shows the values that have been taken into account when choosing the period.

Table 1: Percentage of null values per year (objective variable & mean of all variables)

Year	% Null values Biological Output TOC	Mean % Null values all variables
2017	0.01%	7.91%
2018	0.01%	6.68%
2019	0.05%	6.11%
2020	1.15%	5.06%
2021	7.84%	2.04%

First of all, the structure and the variables of the dataset (2019 samples) is going to be presented (only variables that correspond to the Biological Input, B40, B41, B42, B44, B45, B46 and Biological Output are presented):

- Structure: The data set is composed by 105,002 rows and 35 variables and it contains multiple missing values.
- Variables:
  - Time: Timestamp of the sample taken.
  - (B40—B41—B42—B44—B45—B46) Caudal oxígeno A: Variable that measures the oxygen flow of input 1 of the corresponding pond.
  - (B40—B41—B42—B44—B45—B46) Caudal oxígeno B: Variable that measures the oxygen flow of input 2 of the corresponding pond.
  - (B40—B41—B42—B44—B45—B46—BI) Temperatura: Variable that measures the water temperature of the pond.
  - (B40—B41—B42—B44—B45—B46) Concentración oxígeno disuelto: Variable that measures the oxygen concentration of the corresponding pond.
  - (B40—B41—B42—B44—B45—B46—BI) pH: Variable that measures the pH value of the corresponding pond.

- BI TOC: Variable that measures the organic carbon value in the Biological Input pond.
- BI NT: Variable that measures the nitrogen value in the Biological Input pond.
- BI caudal A: Variable that measures the water flow of input 1 of the Biological Input pond.
- BI caudal B: Variable that measures the water flow of input 2 of the Biological Input pond.
- BO TOC 24H (Target): Variable that measures the organic carbon value in the Biological Output pond 24 hours in the feature.
- BO TOC 48H (Target): Variable that measures the organic carbon value in the Biological Output pond 48 hours in the feature.

All of the above variables are numerical. In addition, the oxygen flow is going to be added in one variable per pond to ease the analysis as it does not influence.

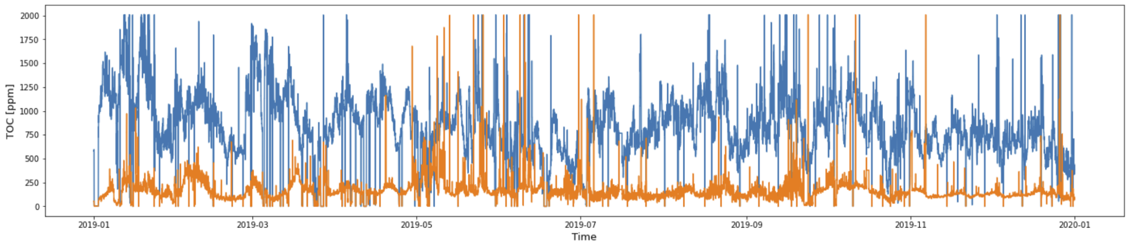


Figure 17: Time series corresponding to biological input and output TOC over the 2019 year, blue and orange respectively.

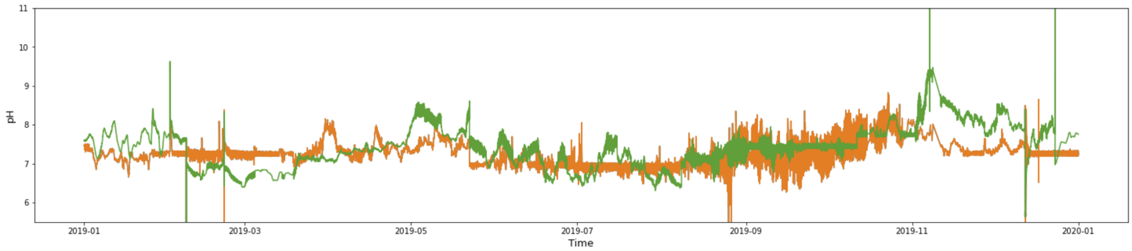


Figure 18: Time series corresponding to B40 and B42 pH's over 2019 year, green and orange respectively

4.3.2 Signal analysis - target variable

In this section, the univariate statistical analysis of the time series corresponding to the target variable, total organic carbon (TOC), is performed. These characteristics allow us to know how the variable behaves over time and establishes the bases for subsequent modeling, as well as the models that will be capable of making predictions of future values.

4.3.2.1 Missing values

As previously mentioned, 2019 has been chosen, among other reasons, for its low content of null values in the target variable. Having 265 null values that corresponds to 0.05% of the total samples. Below is a heat map that allows identifying how these values are located within the timeline.

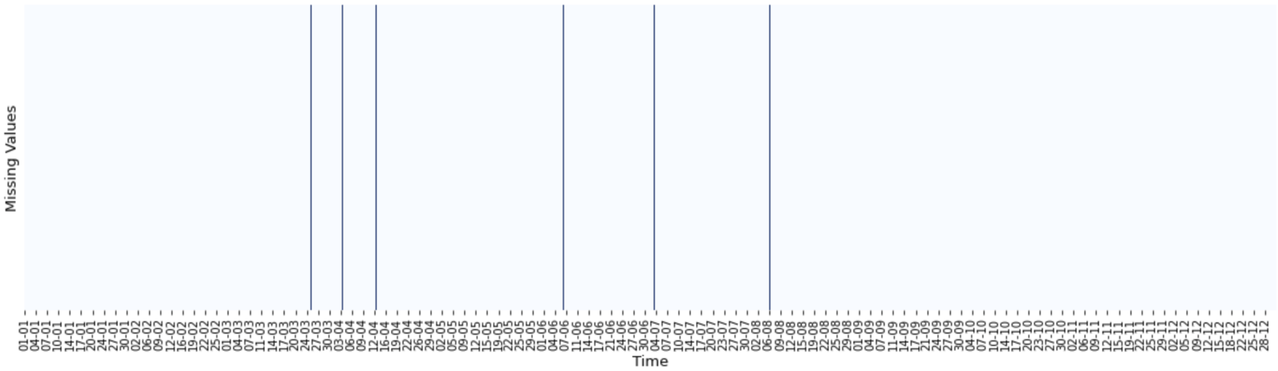


Figure 19: TOC null values heatmap representing the areas of the TOC time series whose values are null. Those areas are represented with dark colors.

As seen in the documentation, there are several options for handling missing values:

- Fill the entries:
  - Fill the missing value with a non-representative value such us inf or zero, depending on the case.
  - Fill the missing value with the mean.
  - Fill the missing value with the last value recorded
  - Fill the missing values with linear interpolate values
- Remove samples or features with missing information
- Select a period of time

The decision has been to fill in the entries and for this the visual results of the methods mentioned above are presented below. As a result, it has been decided to linearly interpolate the missing values.

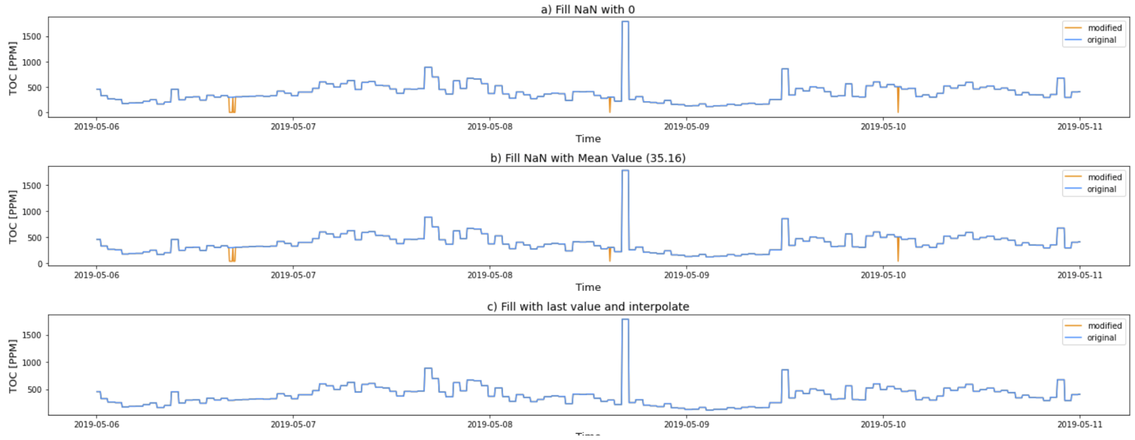


Figure 20: Results for methods used to fill values: a) Fill with 0, b) Fill with the mean value (35,16) and c) fill with last value and interpolate.

4.3.2.2 Frequency and resample

The sampling frequency of the industrial process parameters is 5 minutes. This interval is quite low for the time horizon to be predicted and analyzed (24-48 hours). In addition, the variation of the parameters of the ponds, as well as the organic carbon present, undergo changes very gradually, being processes that change in a matter of hours or days. Therefore, it has been decided to resample the signal to adapt it to the requirements. Resampling must allow it to continue to have sufficient sensitivity to changes to be able to continue detecting variations in the parameters.

The following graph shows the difference in the view of the time series depending on the range chosen for the resample. As can be seen, with the weekly resample a lot of sensitivity to changes is lost, only the trend of the signal is being reflected. Therefore, the applied resampling has been 2 hours, which allows maintaining the sensitivity to changes and obtaining the trend of the signal to predict its behavior, according to the process experts.

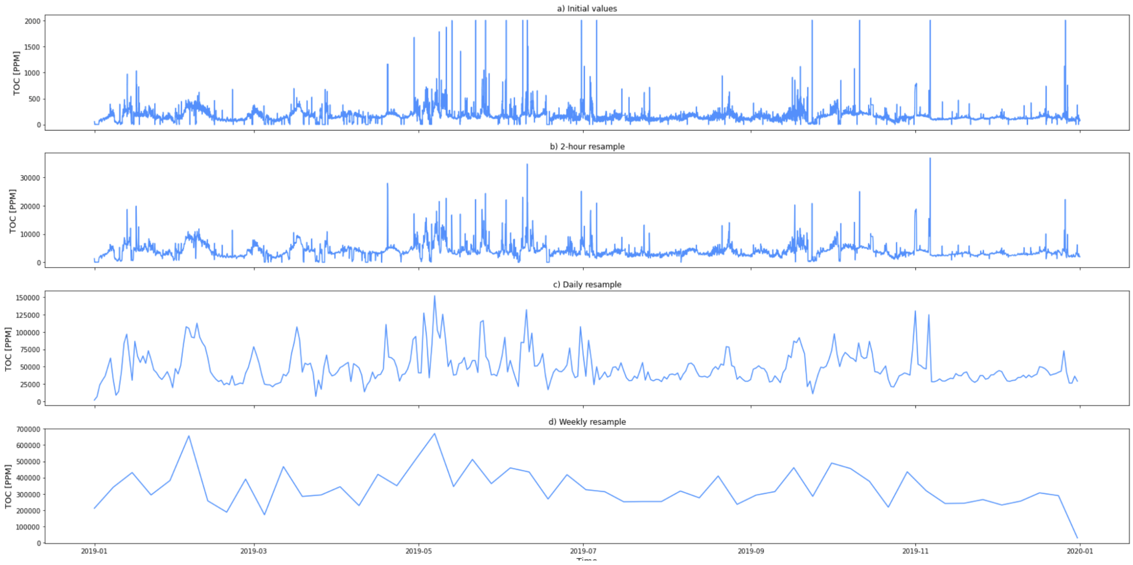


Figure 21: Different kinds of resampling methods for TOC time series: a) Initial TOC values, b) 2-hours resampling, c) Daily resampling and d) Weekly resampling

4.3.2.3 Stationarity

Stationary time-series data is the one whose properties does not depend on the time at which the data is observed, therefore, data with trends or cycles are non-stationary. Some models do assume that data is stationary in order to make predictions. Stationarity can be checked with visual methods or statistical methods.

Stationary data has statistical characteristics like constant mean and constant variance. To check it, the target time series has been plotted together with the 7-days rolling mean and rolling standard deviation. As we can see, nor the mean is constant neither the standard deviation. So that mean that target variable is non-stationary.

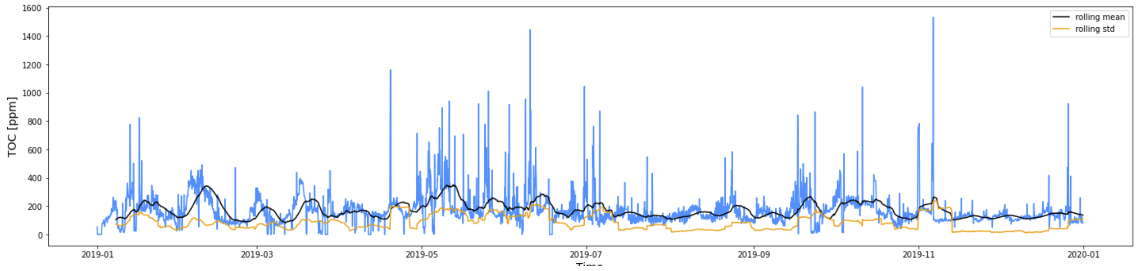


Figure 22: TOC time series and 7-day rolling standard deviation (yellow) and mean (black) over year 2019.

For the statistical analysis, It has been used the Augmented Dickey-Fuller test (ADF) that is a unit root test for time-series data. The Augmented Dickey-Fuller statistic, used in the test, is a negative number. The more negative it is, the stronger the rejection of the null hypothesis that a unit root exists for a certain level of confidence.

- Null hypothesis (N0): Data has a unique root, so data is non-stationary.
- Alternative hypothesis (N1): Time series has no unique root, so data is stationary.

If N0 can be rejected, we conclude that data is stationary. We can reject N0 by two manners. On one hand, if p-value is less than a critical level (usually this level is 5%). On the other hand, by comparing statistics with critical values.

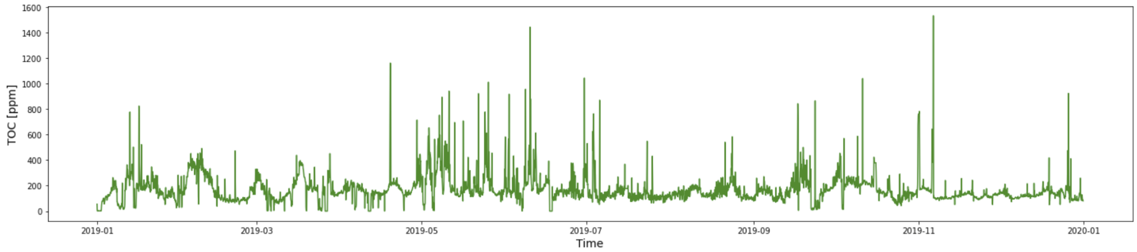


Figure 23: ADF test for TOC variable. ADF Statistic -8.502, p-value: 0.000 — Critical Values 1%: -3.432, 5%: -2.862, 10%: -2.567

As we can see, Null hypothesis (N0) can be rejected, so we can say that data is stationary. P-value is less than 0.05 and ADF Statistic is below 10% threshold.

4.3.2.4 Time series decomposition

Time series decomposition is a technique that is able to split time-series data into several components that represent the real time-series. The components are:

- Level: Average value of the time-series data.
- Seasonality: Describes the periodic signal in the time-series.
- Trend: Describes the direction of the time series, if it is increasing, decreasing or constant.
- Noise: Describes the information left after separating seasonality and trend. Is the variability of the data that cannot be explained.

Decomposition is useful for understanding the time-series data generally and for understanding the nature of each component of the data. Although decomposing the temporal sequence may not be very useful for chaotic signals, we may find some characteristic that indicates a pattern or signal of seasonality, and therefore, deepen the analysis of seasonality.

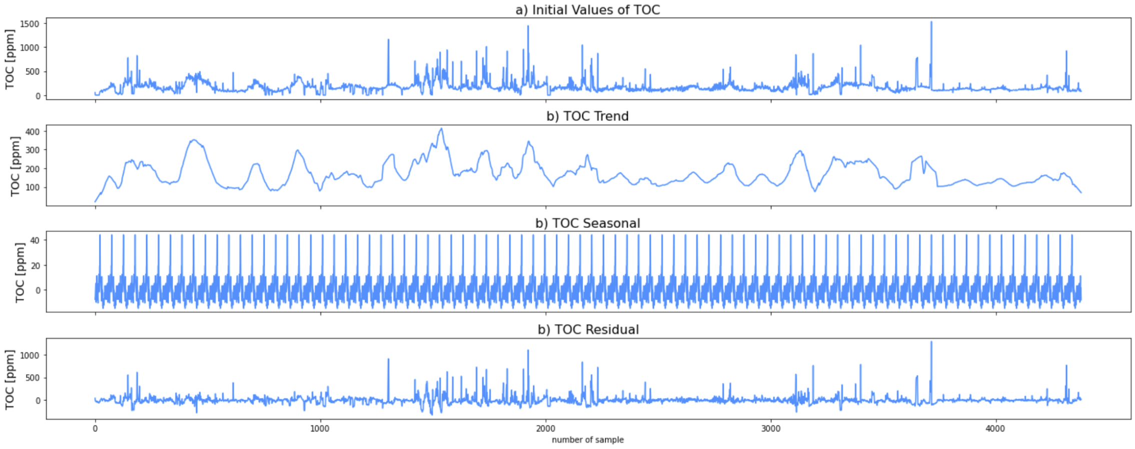


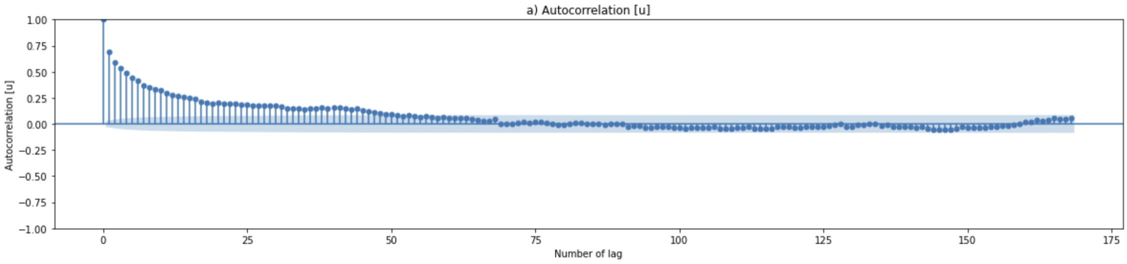
Figure 24: Decomposition of TOC time series: a) Initial TOC values, b) TOc trend, c) TOC seasonal and d) TOC residual.

4.3.2.5 Autocorrelation analysis

In order to check the autocorrelation of time-series data, we are going to use Autocorrelation Function (ACF) and Partial Autocorrelation Function (PACF).

- Autocorrelation Function: Autocorrelation Function (ACF) gives a value that represents how similar a value is within a time-series with a lagged version of it.
- Partial Autocorrelation Function (PACF): A partial autocorrelation is a summary of the relationship between an observation in a time series with observations at prior time steps with the relationships of intervening observations removed.

The correlation and autocorrelation will be reviewed in a time range of two weeks, although the industrial process is of shorter duration and, a priori, there should be no correlation, although the water entering the plant may have some type of temporal correlation that can influence the final correlation of the study variable.



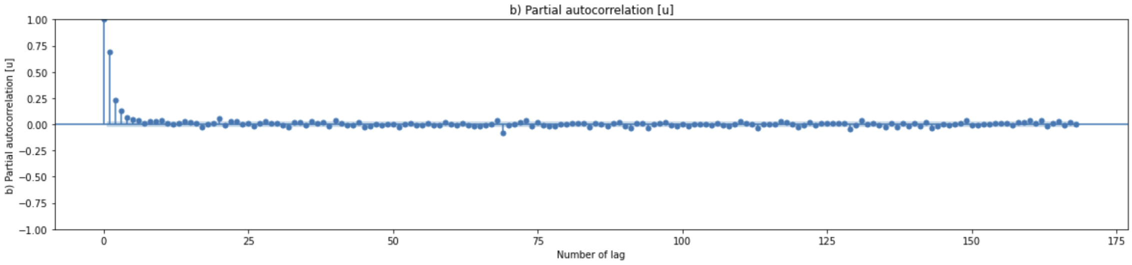


Figure 25: Autoorrelation and Partial autocorrelation of TOC analysis: a) autocorrelation and b) partial autocorrelation (máximum number of lags 168).

We can see that the autocorrelation decreases as the periods advance, there does not seem to be a high autocorrelation beyond the autocorrelation that can exist in a process whose changes are gradual and not abrupt. It seems that there is no periodicity in the inflow of water that could affect the correlations of total organic carbon.

4.3.3 Auxiliary signal analysis

As previously mentioned, the analysis of multivariate series is based on the possible direction and causality of the variables over time. This allows establishing temporal ties that make the models capable of finding patterns that model processes.

Table 2: Number and percentage of missing values per variable for 2019.

Variable	Total Null Values	Percentage Null Values
Oxygen flow B40	112	0.106%
Ph B40	113	0.107%
Soda B40	1068	1.016%
Oxygen flow B41	122	0.116%
Ph B41	113	0.107%
Oxygen flow B42	120	0.114%
Ph B42	128	0.121%
Soda B42	2502	2.380%
Oxygen flow B44	114	0.108%
Ph B44	165	0.156%
Oxygen flow B45	143	0.136%
Ph B45	114	0.108%
Soda B45	133	0.126%
Oxygen flow B46	143	0.136%
Ph B46	126	0.119%
soda B46	2456	2.336%
TOC Biological Input	354	0.336%

First of all, Variable missing values table shows the number and percentage of null values for the year 2019 of all the auxiliary variables considered for the analysis. As can be seen, the number of null values is quite small, with the variables referring to the soda input being the ones with the most null values. Due to the reduced number of null values, interpolation has been performed to fill the dataset.

There are various methods to establish relationships between the variables of a dataset, generic numerical methods and methods focused on the analysis of time series. In this project we will use a combination of both of the knowledge acquired about the industrial process and the information obtained about it.

As a first analysis, a labeling of the TOC target variable has been carried out according to its status:

- $<$ 150: Accepted (Green).
- 150 $\leq$ TOC $<$ 300: High (Orange).
- $\geq$ 300: Critical (Red).

After this labeling, histograms were created for each variable of the ponds: pH, oxygen flow, and, in some cases, soda dosification. This would allow establishing whether for certain values of the auxiliary variables the TOC values were always of the same state, indicating a clear influence on the final TOC value. Below there is a series of histograms:

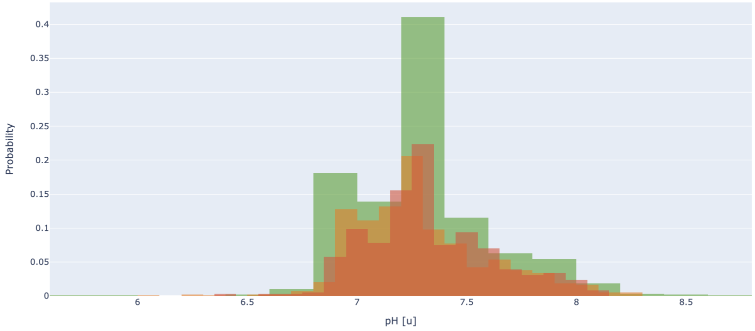


Figure 26: B40 Ph histogram labeled with TOC status: green - low ($TOC < 150$), orange - high ($150 \leq TOC < 300$), red - critical ($TOC \geq 300$).

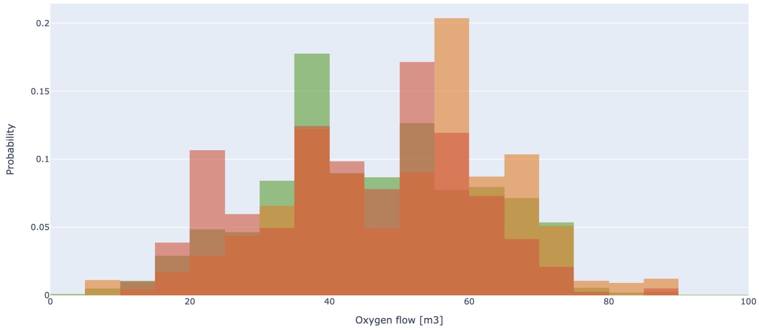


Figure 27: B44 oxygen flow histogram labeled with TOC status: green - low ($TOC < 150$), orange - high ($150 \leq TOC < 300$), red - critical ($TOC \geq 300$).

As can be seen, no clear clustering is obtained. This is repeated for the rest of the variables of the pools, none of which achieves a clear differentiation between the different states of the TOC.

Another experiment that has been carried out has been the calculation of the correlation matrices between the different variables of the ponds and the TOC and CRP metrics. This correlation has been calculated using the Pearson correlation, its equation is presented below:

$$r = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}} \tag{10}$$

where r is the Pearson coefficiente, n represents the sample size, $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$ is the sample mean (analogously for $\bar{y}$). Correlation coefficient, r, tells us how closely data in a scatterplot fall along a straight line. The closer that the absolute value of r is to one, the better that the data are described by a linear equation. If $r = 1$ or $r = -1$ then the data set is perfectly aligned. Data sets with values of r close to zero show little to no straight-line relationship.

If this calculation is performed between each of the variables, a matrix is obtained that can be represented by a heatmap and that offers us a view of the correlations of the dataset. This heat map has been made for each pool, below are those obtained for pools B42 and B46.

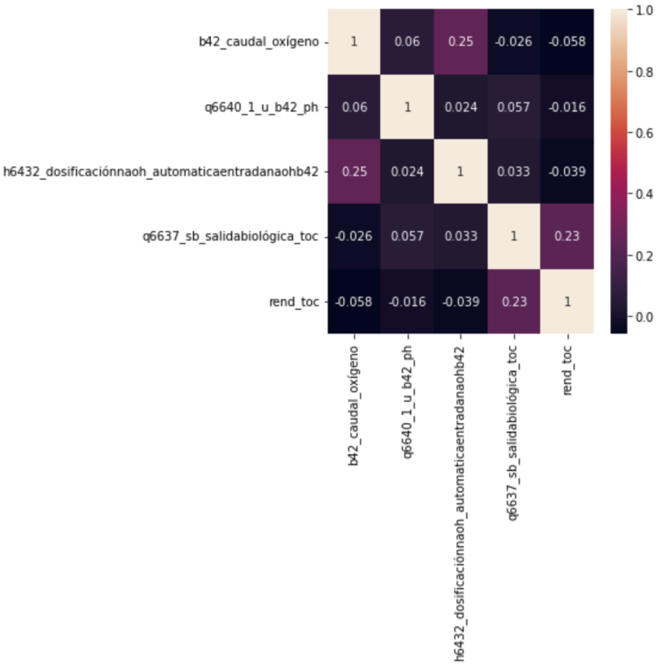


Figure 28: B42 - TOC - CRP variables Pearson's correlation

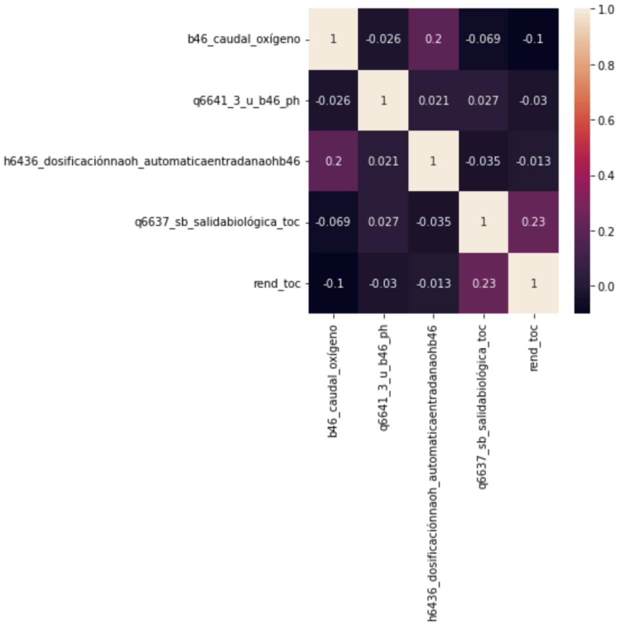


Figure 29: B46 - TOC - CRP variables Pearson's correlation

As can be seen, there is no clear correlation between the variables of the pools with TOC and CRP.

Furthermore, and as discussed in the theoretical part, the Granger Causality test has been applied. The following table shows the results:

Table 3: Granger causality test results

Variable	**6 lags (12 hours)**	**12 lags (1 day)**	**24 lags (2 days)**
Oxygen flow B40	0.77	0.91	0.91
Ph B40	0.95	0.93	0.93
Soda B40	0.53	0.65	0.65
Oxygen flow B41	0.64	0.88	0.88
Ph B41	0.95	0.93	0.93
Oxygen flow B42	0.97	0.81	0.81
Ph B42	0.88	0.92	0.92
Soda B42	0.97	0.98	0.98
Oxygen flow B44	0.82	0.87	0.87
Ph B44	0.01	0.21	0.82
Oxygen flow B45	0.18	0.24	0.24
Ph B45	0.59	0.48	0.48
Soda B45	0.11	0.05	0.05
Oxygen flow B46	0.18	0.24	0.24
Ph B46	0.51	0.87	0.87
soda B46	0.51	0.18	0.18

As can be seen, the test has been carried out with different maximum lags: 12 hours, 1 day and 2 days. Despite this, the results are not conclusive, causality has only been detected in the case of the pH of the B44 pool for 12 hours.

Despite these results, process experts consider that the most critical parameter is pH. This parameter very actively influences how bacteria degrade the carbon present in the water and is the main parameter that makes the population of bacteria increase or decrease.

Therefore, the modeling has been carried out taking into account this information and using only the pH of the ponds, if the modeling is poor, other experiments will be carried out.

4.4 Modelling and evaluation

Once the problem has been described, the analysis of the TOC signal and the analysis of auxiliary variables have been carried out, we can start with the modelling about prediction horizon of 24-48 hours that will allow the operators to anticipate possible erroneous range of TOC in the water. This section will be used to present the modelling parts that have taken place in this project.

4.4.1 Preprocessing

In addition to the actions presented in sections Missing values and Frequency and resample, a series of pre-processing steps have been carried out with the intention of making the modelling as correct as possible.

4.4.1.1 Dataset used

All the variables that do not correspond to the variables that describe the pH's of the ponds have been eliminated with the intention of avoiding the introduction of unnecessary complexity for the model and under the premises explained in section Auxiliary signal analysis. Therefore, the dataset used for modelling consists of 9 variables: 6 corresponding to the measurements of the pH of the ponds, the initial value of the TOC in the biological input and two target variables that correspond to the TOC 24h and 48h later measured in the biological output.

Table 4: Dataset used for modelling with Min, Max and Mean statistics for all variables.

Variable	Type	Max	Min	Mean
Ph B40	Numerical — Auxiliary	5.64	8.36	7.29
Ph B41	Numerical — Auxiliary	5.64	8.63	7.29
Ph B42	Numerical — Auxiliary	6.13	9.40	7.44
Ph B44	Numerical — Auxiliary	4.30	9.33	7.54
Ph B45	Numerical — Auxiliary	5.07	9.60	7.38
Ph B46	Numerical — Auxiliary	5.34	9.69	7.63
Biological Input TOC	Numerical — Auxiliary	639.71	1942.22	843.77
Biological Output TOC 24h	Numerical — target	112.65	1533.23	170.87
Biological Output TOC 48h	Numerical — target	112.65	1533.23	170.87

4.4.1.2 Scaling

Some of the models (MLP and SVR), used in this phase, require data to be scaled. There are several methods to normalize data and each has its advantages and disadvantages. In this case, the Min-Max rescaling is used because it is useful when trying to compare datasets of different factors or that use different units (pH and TOC in ppm's).

4.4.1.3 Splitting and shuffling data

For all the experiments described later, a division has been made in the dataset as follows: 70% for training and 30% for testing. With the data available for modelling, the number of samples dedicated to training and testing is 2971 and 1274, respectively. This allows validating the prediction for both the model with vision at 24h and the model with vision at 48h.

The training data has been shuffled before being introduced into the model, but in the test set the order has not been modified to correctly observe the visual results with the intention of validating the proactivity of the model before the TOC trend changes. This will help to achieve a more robust model and will help to validate the experiments.

Table 5: Split Train & Test set

Set	Initial time	End time	Number of samples
Train	01-01-2019 00:00:00	07-09-2019 00:00:00	2971
Test	07-09-2019 02:00:00	29-12-2019 20:00:00	1274

4.4.2 Metrics

The metrics used to compare the performance between models are Mean Squared Error (MSE) and R2 Score or coefficient of determination.

- MSE: Mean square error (MSE) is the average of the square of the errors. The larger the number, the larger the error. Error in this case means the difference between the observed values x_1, x_2, x_3, . . . , x_n, and the predicted ones y_1, y_2, y_3, . . . , y_n. We square each difference $(x_i – y_i)^2$ so that negative and positive values do not cancel each other out.

$$MSE = \frac{1}{n}\sum_{i=1}^{n}(x_i - y_i)^2 \tag{11}$$

  where n is the sample size, x_i is the observed value at position i and y is the predicted value at position i.

- R2 Score: R2, the coefficient of determination, determines the ability of a model to predict future outcomes. The best possible result is 1.0, and it occurs when the prediction matches the values of the target variable. R2 can take negative values because the prediction can be arbitrarily bad. When the prediction matches the expected values of the target variable, the result of R2 is 0. It is defined as 1 minus the total sum of squares divided by the sum of squares of the residuals.

$$R^2 = 1 - \frac{sum\_squared\_regression(SSR)}{total\_sum\_of\_squares(SST)}, 1 - \frac{\sum(y_i - \hat{y}_i)^2}{\sum(y_i - \overline{y})^2} \tag{12}$$

  where y_i is the observed value at position i, $\hat{y}_i$ is the predicted value at position i and $\overline{y}$ is the mean of the observed values.The sum squared regression is the sum of the residuals squared, and the total sum of squares is the sum of the distance the data is away from the mean all squared.

4.4.3 Models used and model tunning

There are simple models that can be surprisingly effective for time series analysis tasks. Three different models have been used for this project: Multi-Layer Perceptron Regressor (MLP), Support Vector Regressor (SVR) and XGBoost Regressor, all obviously dedicated to regression tasks. The theory behind these models has been explained in section Modelling and evaluation.

To improve its performance, a search of hyperparameters has been carried out in order to find the ones that offer the best performance for the characteristics of the available data. Below is a description of the hyperparameters for each of the models, the tested values and the final values used in the experiments. The search for hyperparameters has been carried out using the Stratified Cross-Validation technique with 3 folds on the training set.

- MLP Regressor:
  - Hidden Layer Sizes: It represents the number of hidden layers and the number of neurons per layer.
  - Activation Function: Activation function for each hidden layer.
    - Tanh: the hyperbolic tan function, returns: f(x) = tanh(x).

      $$f(x) = tanh(x) \tag{13}$$

    - Relu: the rectified linear unit function, returns:

      $$f(x) = max(0, x) \tag{14}$$

  - Solver: Solver for weight optimization.
    - SGD: refers to stochastic gradient descent.
    - ADAM: refers to a stochastic gradient-based optimizer proposed by Kingma, Diederik, and Jimmy Ba.
- SVR:
  - Kernel: Specifies the kernel type to be used in the algorithm.
    - RBF: Radial-Basis Function Kernel

      $$f(x) = \sum_{i=1}^{n} \alpha_i g(x - x_i). \tag{15}$$

  - C: Regularization parameter, which inverse proportional to C.
  - Epsilon: It specifies the epsilon-tube within which no penalty is associated in the training loss function with points predicted within a distance epsilon from the actual value.

- XGBOOST Regressor:
  - Alpha: L1 regularization term on weights. Increasing this value will make model more conservative.
  - Lambda: L2 regularization term on weights. Increasing this value will make model more conservative.
  - Estimators: Number of estimators to be used in the model.
  - Maximum depth: Maximum depth of a tree. Increasing this value will make the model more complex and more likely to overfit. 0 indicates no limit on depth.
  - Learning Rate: Step size shrinkage used in update to prevents overfitting.

The tests have been carried out with parameters obtained through an iterative process of trial and error based on other configurations in the literature and the error results obtained. For SVR [27], XGBOOST Regressor [28], MLP Regressor [29].

Table 6: Hyperparameter searching XGBOOST

Parameter	**V1**	**V2**	**V3**
Alpha	0.1	0.01	
Lambda	0.001	0.1	
Estimators	100	500	1000
Maximum depth	0	6	10
Learning rate	0.01	0.1	

Table 7: Hyperparameter searching SVR

Parameter	**V1**	**V2**	**V3**	**V4**	**V5**
C	200	100	10	1	0.1
Epsilon	1	0.1	0.01	0.001	0.0001
Kernel	RBF				

Table 8: Hyperparameter searching MLP

Parameter	**V1**	**V2**	**V3**
Hidden layer sizes	(10, 10)	(5, 5)	(20, 20)
Activation function	relu	tanh	
Solver	adam	sgd	

Here there are presented the best parameters for each of the models obtained, 24 hours and 48 hours prediction models:

- 24 Hours:

Table 9: Best parameters XGBOOST 24h

Parameter	**Best Val**
Alpha	0.1
Lambda	0.001
Estimators	500
Maximum depth	10
Learning rate	0.1

Table 10: Best parameters SVR 24h

Parameter	**Best val V5**
C	200
Epsilon	0.01
Kernel	RBF

Table 11: Best parameters MLP 24h

Parameter	**Best Val**
Hidden layer sizes	(20, 20)
Activation function	relu
Solver	adam

- 48 hours:

Table 12: Best parameters XGBOOST 48h

Parameter	**Best Val**
Alpha	0.1
Lambda	0.001
Estimators	500
Maximum depth	10
Learning rate	0.1

Table 13: Best parameters SVR 48h

Parameter	**Best val V5**
C	200
Epsilon	0.01
Kernel	RBF

Table 14: Best parameters MLP 48h

Parameter	Best Val
Hidden layer sizes	(5, 5)
Activation function	relu
Solver	adam

4.4.4 Results

This section describes the results based on the conditions explained above, with special emphasis on the metrics described. Comparison plots are shown that allow understanding the performance of each of the models used, likewise, time series graphs that allow observing the predictions made on the test set to have a vision of how the model has responded to data never seen before.

4.4.4.1 Forecasting results: 24h

The results for the models trained for a 24-hour forecast are presented below. A comparison is shown based on MSE and R2 Score. In addition, a temporary graph is shown over the entire test set and a zoom to be able to observe in more detail the proactivity of each of the models.

Multi-Layer Perceptron

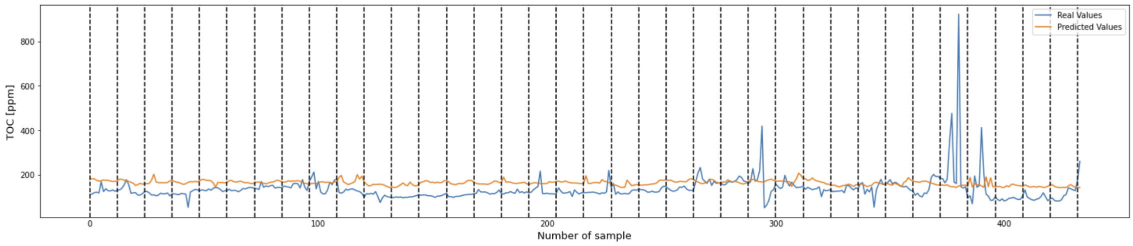


Figure 30: Predictions 500 samples test set 24h: real values (blue) and predicted values MLP Regressor (orange).

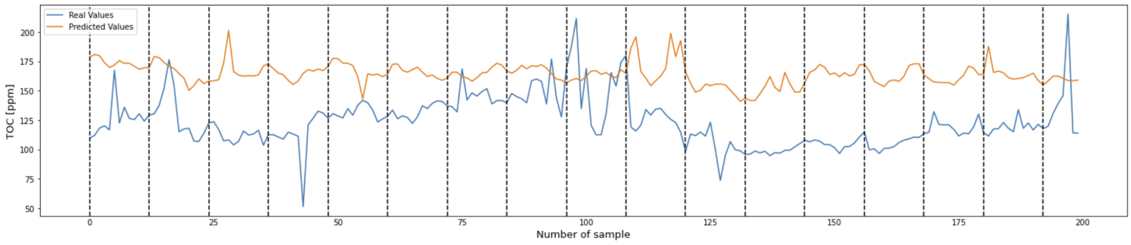


Figure 31: Predictions 200 samples test set 24h: real values (blue) and predicted values MLP Regressor (orange).

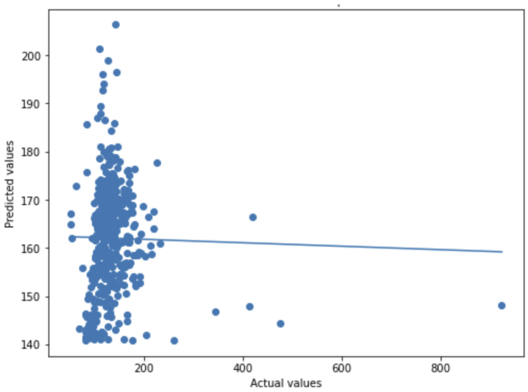


Figure 32: R2 score plot MLP Regressor: Coefficient of determination: -0.31901, MSE: 3973.99

As we can see in the results corresponding to the MLP regressor, the model is not able to recognize the patterns within the data, not being able to observe the delay and the modifications between the TOC values of the biological input and output. If we carefully observe the temporal graph, we can see that the signal is replicated one day apart, indicating a transposition of the signal at t+24h. In addition to the aforementioned transposition, there is a vertical one, which further distances the results from reality. The lack of concordance of the predictions with reality is reflected in the R2 Score graph, with the value below 0, indicating that the prediction is worse than the random prediction.

Support Vector Regressor

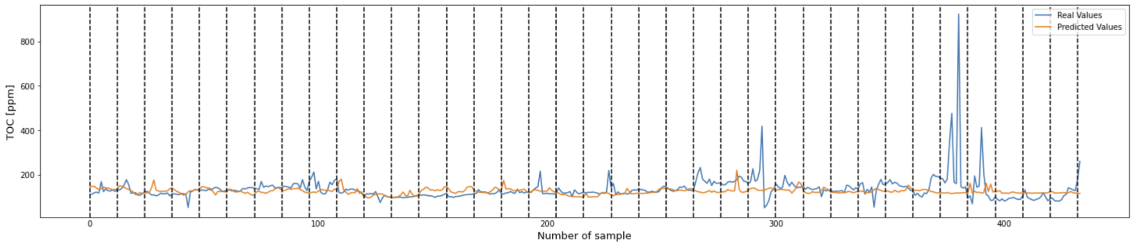


Figure 33: Predictions 500 test set 24h: real values (blue) and predicted values SVR (orange).

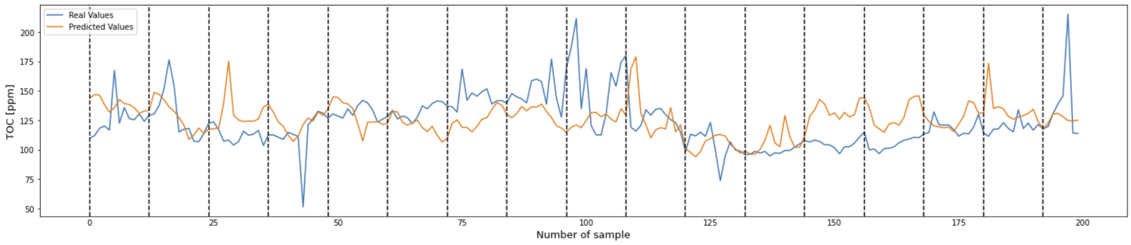


Figure 34: Predictions 200 samples test set 24h: real values (blue) and predicted values SVR (orange).

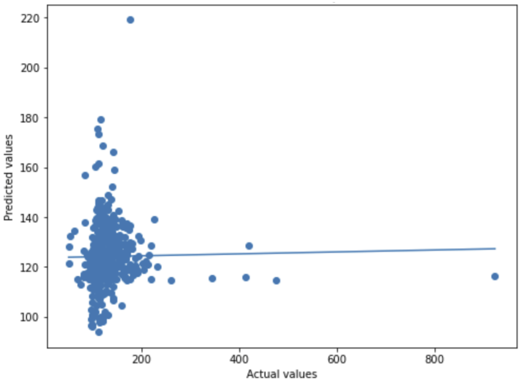


Figure 35: R2 score plot SVR: Coefficient of determination: -0.07773, MSE: 3247.06

The results for the SVR are quite similar to those obtained for the MLP regressor. As can be seen, it is also unable to interpret the delay and changes that exist between the input biological TOC and the output biological TOC. This is evident by looking at the transposition that exists from t+24h. Both the R2 Score and the MSE are pretty bad so the results are not good.

XGBOOST Regressor

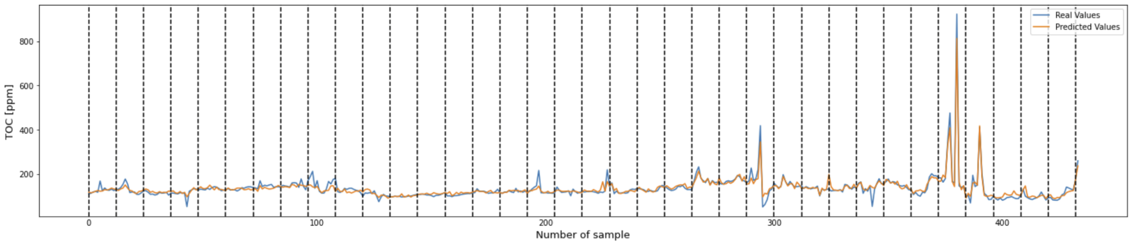


Figure 36: Predictions 500 samples test set 24h: real values (blue) and predicted values XGBOOST Regressor (orange).

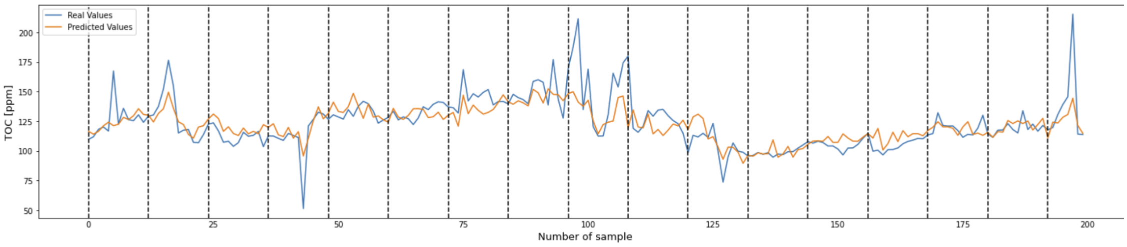


Figure 37: Predictions 200 samples test set 24h: real values (blue) and predicted values XGBOOST Regressor (orange).

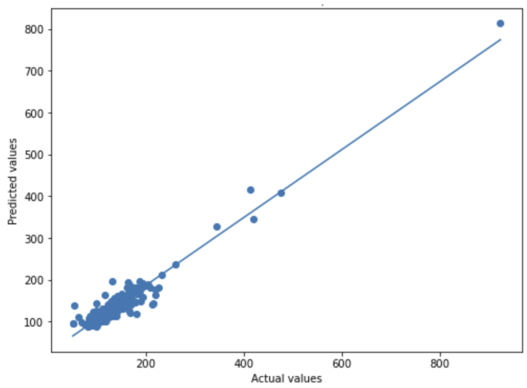


Figure 38: R2 score plot XGBOOST Regressor: Coefficient of determination: 0.91375, MSE: 259.86

In this case the results change quite a bit with respect to the previous two. It can be seen that the interpretability of the model is quite good, the delay is almost non-existent and the model acts proactively before of changes in the TOC trend. Both in the general view and in the extended view of the time series, it can be seen that the model predicts with some accuracy the real changes of the test set. Also, the R2 Score graph is very good, the linear regression is very correct and the value is very close to 1.

Forecasting models comparison: 24h

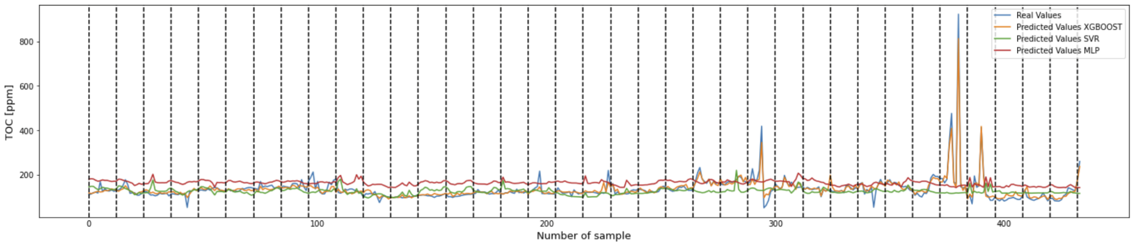


Figure 39: Predictions 500 samples test set 24h: real values (blue), predicted values XGBOOST Regressor (orange), predicted values SVR (green) and predicted values MLP Regressor (red).

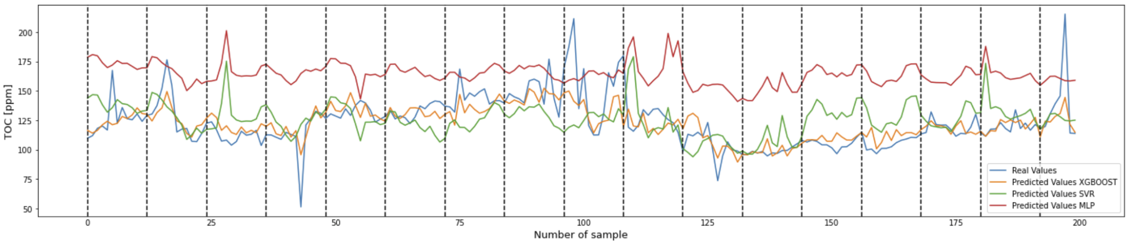


Figure 40: Predictions 200 samples test set 24h: : real values (blue), predicted values XGBOOST Regressor (orange), predicted values SVR (green) and predicted values MLP Regressor (red).

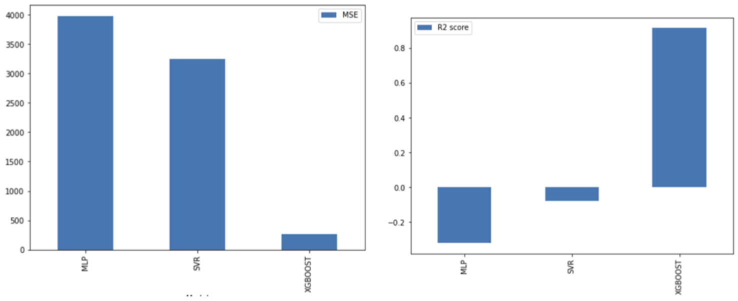


Figure 41: MSE (left plot): MLP Regressor, SVR and XGBOOST Regressor (3973.99, 3247.06, 259.86) — R2 score (right plot): MLP Regressor, SVR and XGBOOST Regressor (-0.31090, -0.07773, 0.91375).

In reference to the comparison of results between models, we can observe what was previously mentioned. The model that best responds to the temporal patterns of the data is XGBOOST, whose metrics are very good. As for MLP Regressor and SVR, it can be seen that both models are penalized by the observed transpositions, with MLP Regressor being the most penalized model due to its two transpositions, both vertical and horizontal.

4.4.4.2 Forecasting results: 48h

The results for the models trained for a 48-hour forecast are presented below. A comparison is shown based on MSE and R2 Score. In addition, a temporary graph is shown over the entire test set and a zoom to be able to observe in more detail the proactivity of each of the models. The results for the prediction of 48 hours ago are quite similar to those obtained for 24 hours, the results are detailed below.

Multi-Layer Perceptron

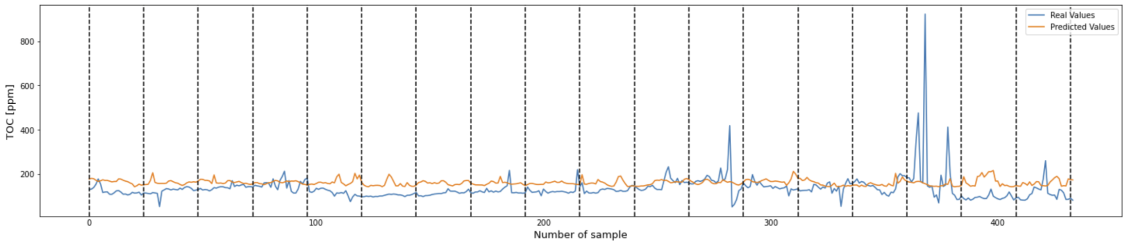


Figure 42: Predictions 500 samples test set 48h: real values (blue) and predicted values MLP Regressor (orange).

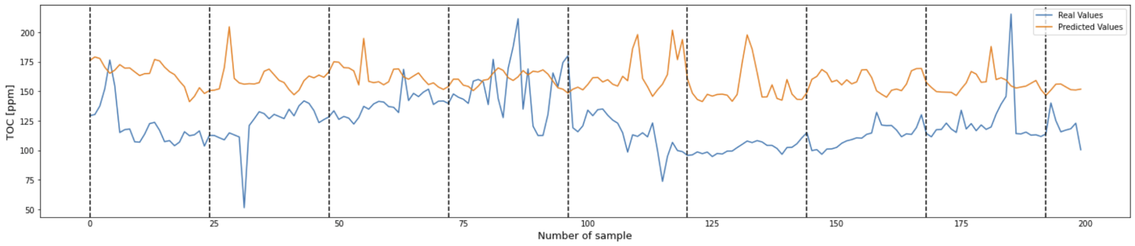


Figure 43: Predictions 200 samples test set 48h: real values (blue) and predicted values MLP Regressor (orange).

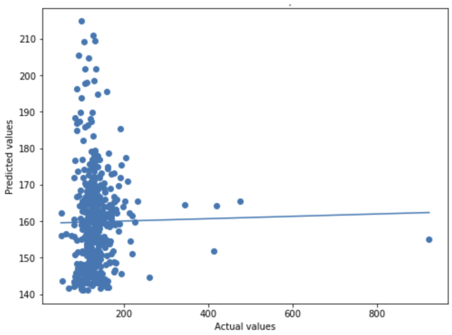


Figure 44: R2 score plot MLP Regressor: Coefficient of determination: -0.29152, MSE: 3928.02

The results obtained for the MLP Regressor are quite similar to those obtained for 24 hours. The model is not able to find the inherent patterns in the data and therefore the same horizontal and vertical translation is obtained, leading to very poor results.

Support Vector Regressor

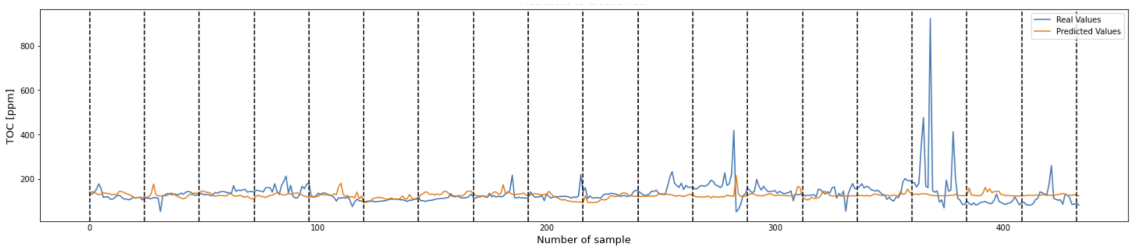


Figure 45: Predictions 500 samples test set 48h: real values (blue) and predicted values SVR (orange).

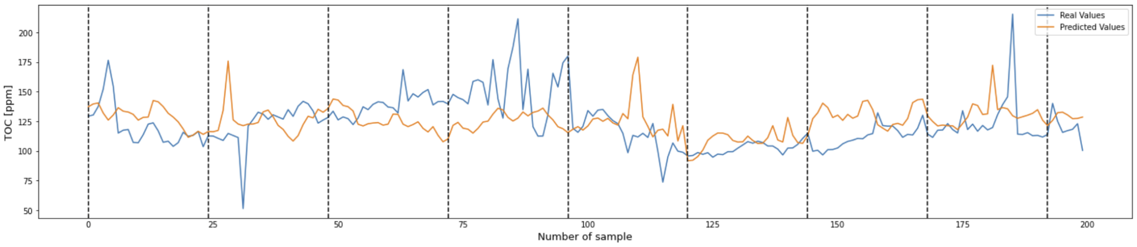


Figure 46: Predictions 200 samples test set 48h: real values (blue) and predicted values SVR (orange).

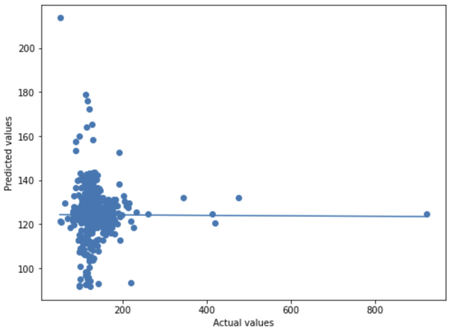


Figure 47: R2 score plot SVR: Coefficient of determination: -0.07637, MSE: 3273.66

The results obtained by the SVR are quite similar to those obtained by the MLP regressor and coincide with the same problems seen in the 24-hour forecast. Although they do not present the vertical translation seen in the MLP Regressor, the horizontal translation is maintained, which prevents good results from being obtained.

XGBOOST Regressor

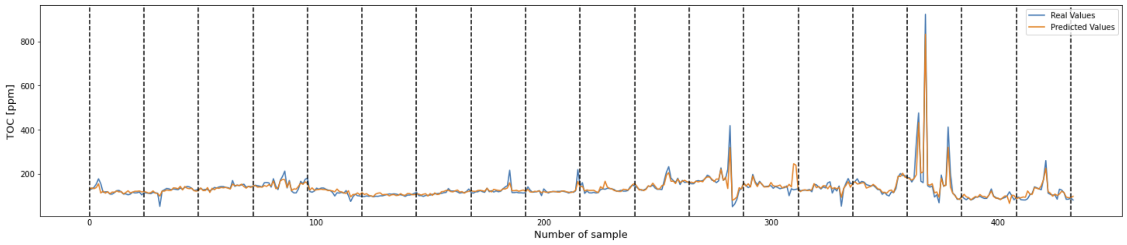


Figure 48: Predictions 500 samples test set 48h: real values (blue) and predicted values XGBOOST Regressor (orange).

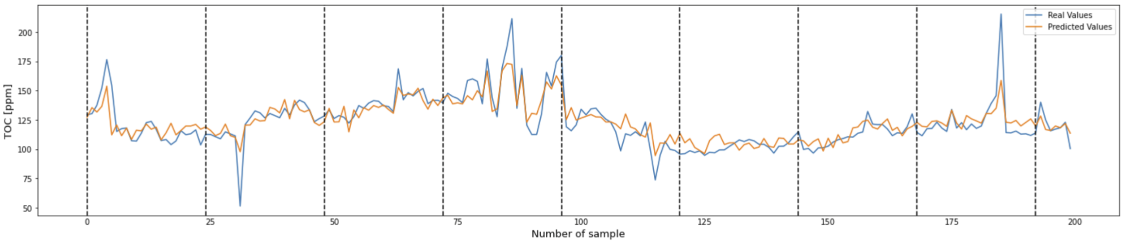


Figure 49: Predictions 200 samples test set 48h: real values (blue) and predicted values XGBOOST Regressor (orange).

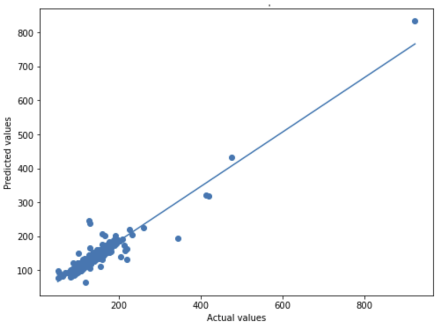


Figure 50: R2 score plot XGBOOST Regressor: Coefficient of determination: 0.8903, MSE: 9.41843

The predictions obtained by the XGBOOST Regressor are still the best, and its results are good. The interpretability of the offset is very good, being able to obtain results that are very close to reality. In addition, the determination coefficient continues to be high, not being penalized by the increase in the prediction window to 48 hours. The model's proactivity in anticipating trend changes is good.

Forecasting models comparison: 48h

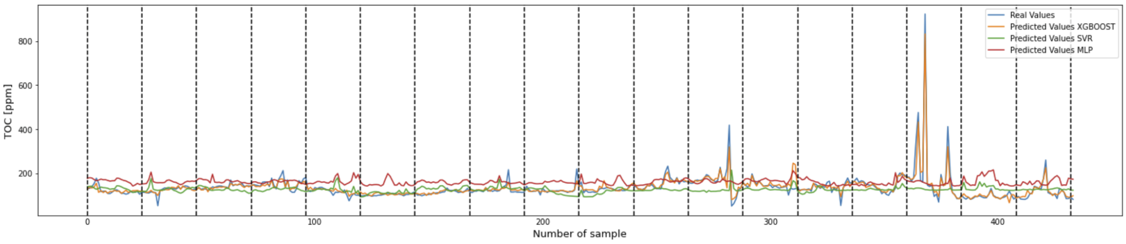


Figure 51: Predictions 500 samples test set 48h: real values (blue), predicted values XGBOOST Regressor (orange), predicted values SVR (green) and predicted values MLP Regressor (red)

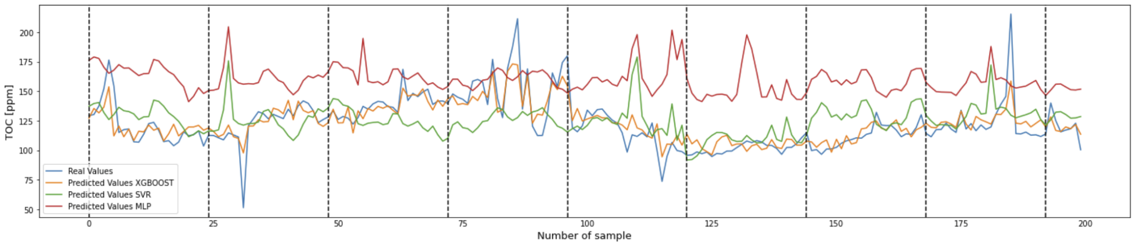


Figure 52: Predictions 200 samples test set 48h: real values (blue), predicted values XGBOOST Regressor (orange), predicted values SVR (green) and predicted values MLP Regressor (red)

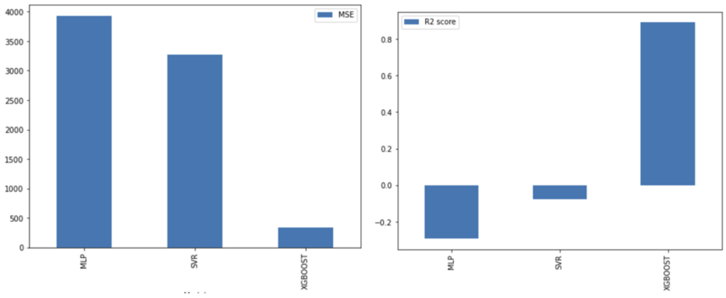


Figure 53: MSE (left plot): MLP Regressor, SVR and XGBOOST Regressor (3928.02, 3273.66, 9.41843) — R2 score (right plot): MLP Regressor, SVR and XGBOOST Regressor (-0.29152, -0.07637, 0.8903).

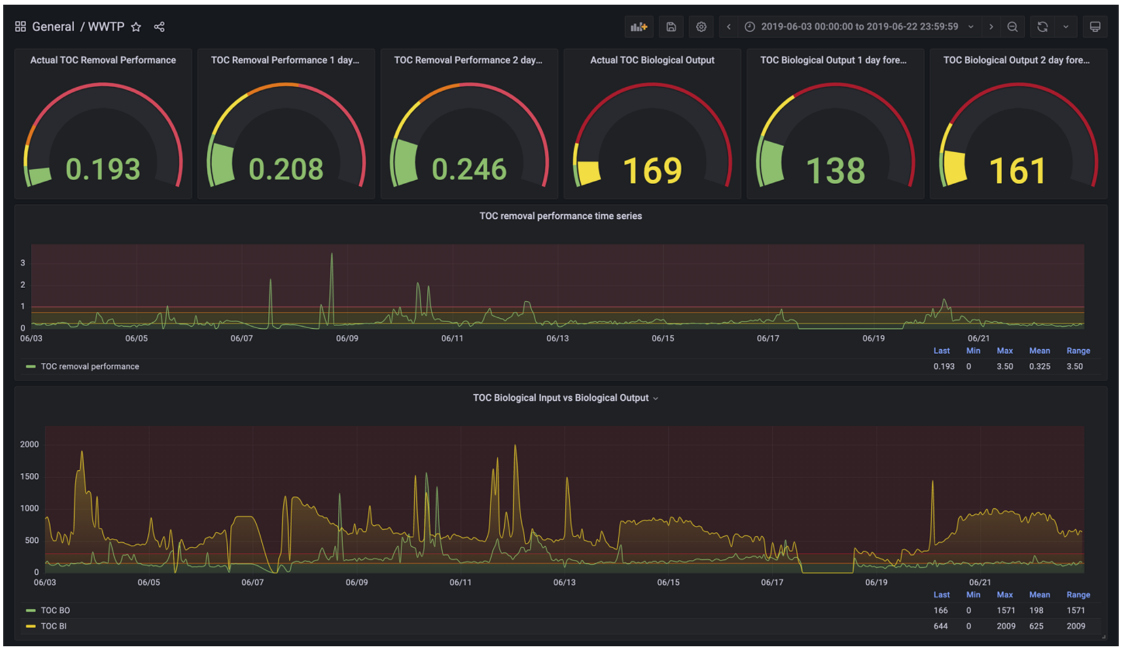
The comparison between models is still quite similar. XGBOOST Regressor improves all the results obtained by the other models. Obtaining both a better R2 Score metric and a better MSE.

4.5 Visualization and deployment

4.5.1 Visualization

To carry out the visualization, a dashboard has been created by integrating InfluxDB with Grafana. This combination has made it possible to create a control panel with different metrics that characterize the industrial process. In the dashboard you can see different indicators with a specific function for each one. The dashboard has been divided into two parts: (1) operational dashboard that has the objective of indicating the most relevant parameters of the plant, being able to see the current values of the indicators to know what their status is, (2) business intelligence dashboard where the predictive analytics is located and for which the models developed in the previous section are used.

The implemented dashboards are described below, with their purpose, graphics and operation of use that allow understanding how they should be used to carry out a correct operation.


General / WWTP
2019-06-03 00:00:00 to 2019-06-22 23:59:59
Actual TOC Removal Performance
0.193
TOC Removal Performance 1 day...
0.208
TOC Removal Performance 2 day...
0.246
Actual TOC Biological Output
169
TOC Biological Output 1 day fore...
138
TOC Biological Output 2 day fore...
161
TOC removal performance time series
3
2
1
0
06/03
06/05
06/07
06/09
06/11
06/13
06/15
06/17
06/19
06/21
TOC Biological Input vs Biological Output
2000
1500
1000
500
0
06/03
06/05
06/07
06/09
06/11
06/13
06/15
06/17
06/19
06/21

	Last	Min	Max	Mean	Range
TOC removal performance	0.193	0	3.50	0.325	3.50

	Last	Min	Max	Mean	Range
TOC BO	166	0	1571	198	1571
TOC BI	644	0	2009	625	2009

Figure 54: Decision Panel from Grafana Dashboard

In the first place we can observe the predictive analytics dashboard. This dashboard shows the indicators related to key metrics such as TOC and organic carbon removal performance (CRP), these are the two metrics that guide the operation of the plant, so it is vitally important to obtain a clear vision and to be able to anticipate to trend changes.

Two well-differentiated parts can be seen. The first part consists of indicators in gauge format, they indicate the current state of the carbon degradation performance and the current value of the TOC. In addition, by means of a color code, the states of the possible ranges are shown, which are:

- For TOC values:
    - TOC < 150: Accepted (Green).
    - $150 \leq$ TOC < 300: High (Orange).
    - TOC ≥ 300: Critical (Red).
- For Carbon Removal Performance (CRP):
    - CRP < 0.25: Accepted (Green).
    - $0.25 \leq$ CRP < 0.75: High (Orange).
    - CRP ≥ 0.75: Critical (Red).

In this first part there is also predictive analytics and it is also shown in gauge format. 4 predictive indicators are displayed: TOC value 24h, TOC value 48h, carbon removal performance 24h and carbon removal performance 48h. These indicators allow anticipating possible upward trends to carry out a more effective pre-treatment of the water, and therefore ensure the correct functioning of the water line. These indicators maintain the same color code as mentioned above.

The second part consists of the historical time series where the evolution of the values of the key metrics, TOC and carbon removal performance, can be observed in detail. Especially, it helps to locate episodes of bad operations. The same color code is also maintained for the indication of the different status of the operation, these thresholds are shown by horizontal lines in the graphs.

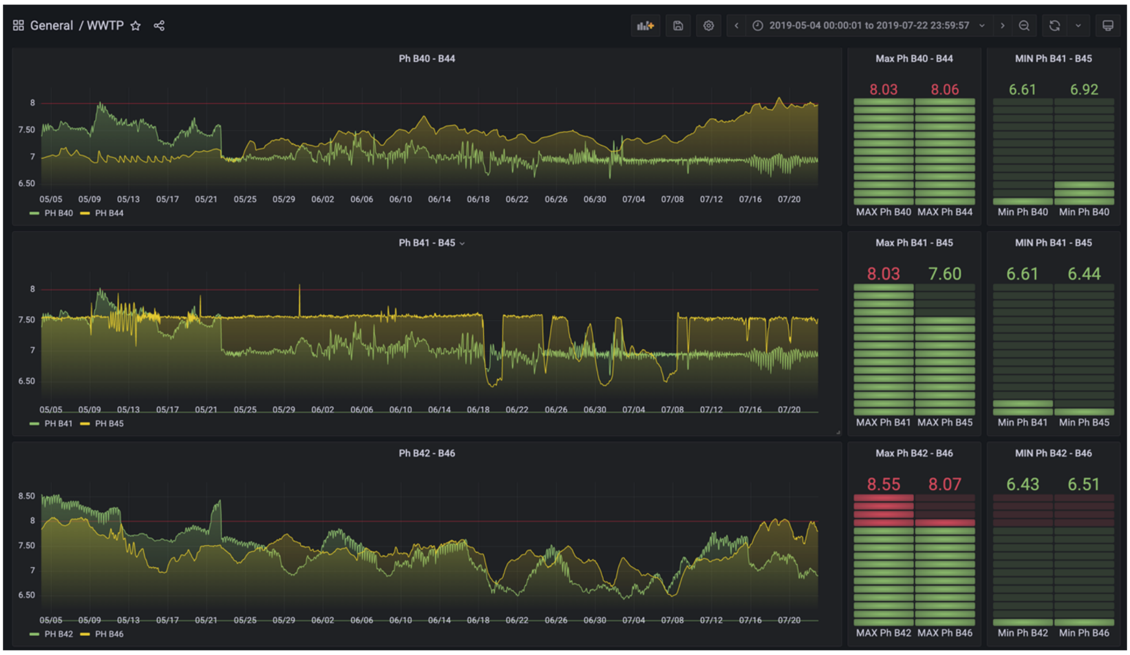


Figure 55: Operational Panel from Grafana Dashboard

The second part of the dashboard consists of the operational part of the process. As we have seen in the section on auxiliary signal variable analysis, the parameters that most influence the operation of the plant is the pH of the ponds, so this variable takes on a special role in the operational part.

For this part, the variables have been divided into three parts, corresponding to the time period of each of them. In the first two graphs we obtain information on the historical, minimum and maximum of the selected period of the Ph of the B40 and B44 ponds.

The two subsequent graphs contain the same information but from the corresponding pools, first pools B41 and B45 and then pools B42 and B46.

Indicators have been defined by means of colors according to the state of the pH of the ponds and their interpretation by the operational, the defined states are detailed below:

- pH < 6: Low level (Red).
- $6 \leq$ pH < 8: Accepted level (Green).
- pH ≥ 8: High level (Red).

This color legend has been established based on the performance criteria of carbon degradation of bacteria, being the range of 6-8 the indicated so that the colony of bacteria obtains the optimal conditions to avoid a decrease of individuals below the range putting at risk the correct degradation of organic carbon. Additionally, in combination with the predictive dashboard, relationships can be established between pH variation and current and future TOC and CRP levels.

4.5.2 Deployment

In theoretical section Deployment, the steps for the deployment of the project and its closure have been established. Therefore, based on the points discussed in this section, we proceed to detail the steps that should be carried out for the integration of the implementation of the practical part in production.

As part of the final report, this document in which all the implementations are collected, theoretical as well as practical, it also contains all the necessary information to know the reasons and the scope of the project.

For the production of the model, an implementation using docker containers is suggested. Docker is a software available for Linux, Windows and MacOS, which packages applications and facilitates their deployment in any environment, its base is the use of lightweight containers that run processes independently. Its advantages are:

- Docker facilitates distribution: These Dockerfiles are very easy to distribute (it is a mere text file, very light), and with them we can build each container from scratch, so dispensing and implementing it is as simple as it is elegant, also relieving the load when distributing software
- Docker deployment is very fast: This is because the containers are optimized: each container has its own library and its own executable, that is, strictly what is neces-

sary. While a virtual machine takes several minutes to launch each container, with Docker we can do it in seconds.

- The layering system in Docker is very optimal: The layered file system that prevails in Docker allows optimizing those that are common to several containers: if different processes require the same file, both are not executed separately, but rather it is served simultaneously.
- Docker makes the most of the server: Thanks to the hyper-specialization of the Docker layers, it is possible to increase the density of each of the servers, taking advantage of resources and optimizing spending, without the loading speed suffering at any time.

As a service infrastructure, the use of cloud platforms is recommended. It enables experimenting and testing multiple models, cloud allows you to scale your machine learning projects up and down as needed. You can start with a small set of data points and add more as you get more confident in your predictions. Traditional machine learning isn't just complex and hard to set up: It's pricey. If you want to train and deploy large machine learning models, such as deep learning, on your own servers, you'll need expensive GPU cards. In order to scale your models to accommodate large-scale needs, you'll need high-end GPU units, which means that they'll remain largely unused during periods of low use. In other words, you'll have expensive servers sitting around collecting dust, while still requiring extensive maintenance. Most popular cloud services also provide SDKs (software developer kits) and APIs. This allows you to embed machine learning functionality directly into applications. They also support most programming languages.

5 Conclusion and future work

5.1 Conclusions

In this project, a methodology for the implementation of analytical applications in the field of Industrial Internet of Things has been introduced, explaining the particularities that this type of applications has. The methodology based on CRIPS-DM adapted to the problems of the IIOT has served to lay the foundations for the practical implementation presented and has guided its implementation.

As can be seen throughout this report, all the points of the proposed methodology have been developed, proposing a theoretical implementation and then using it in a real project to obtain a specific solution. The only points that have not been practically developed are the implementation of an infrastructure dedicated to obtaining data from industrial processes through IoT due to the available access to the extracted data, and it is also outside the scope of the project. On the other hand, some of the parts that correspond to the deployment have been carried out in this report.

In the technical part, highlight three tools that have improved the quality of the practical solution and have helped to obtain a more complete solution:

- InfluxDB is an ideal tool for storing time series, standing out for its efficiency and speed of reading and writing. Its direct integration with IIoT infrastructure management tools, visualization tools and the great support provided by the community make it a very competitive tool.
- After the comparison made between the Multi-Layer Perceptron, Support Vector Regressor and XGBOOST Regressor models, the comparison of the results made was surprising. XGBOOT Regressor has offered very good results compared to the other models, being able to identify the lag problem. The proactivity and anticipation of the model is good and has allowed obtaining the results to present them in the visualization tool.
- Grafana has made it possible to implement visualization quickly. The integration with InfluxDB is direct and very fast, the interface is comfortable to use and it is pleasant offering a really good experience. The tool helps us study, analyze and monitor data over a period of time, technically called time series analysis. It helps us to track the behaviour of key parameters and metrics; the frequency of errors; the type of errors that appear and the contextual scenarios when providing relative data.

5.2 Future work

Data science applied to the Industrial Internet of Things is a novelty in the field of data analytics and, although significant advances have been made in recent years, it is a fairly new field with projections to become a standard. in the industry.

The attempt to characterize chaotic time series has been developing over the last few years.

In this project it is very relevant and, therefore, one possibility to expand this project is to use the latest advances in this field such as ANFIS (adaptive network based fuzzy inference system) models, which are complex models that can model chaotic processes with more accuracy. In addition, the use of data windowing could help to obtain better results in the analysis of chaotic time series, expanding the possibilities of the project.

One of the greatest limitations has been the inability to directly access the IoT devices that collected the data from the industrial process. This has prevented the initial validation of the data; in addition, the quality of these data is not the best since by not having access to the sensors there are many problems in collecting data from them. It has prevented the implementation of the complete flow of the proposed methodology and, although it is not project-based, it would have been important to carry out the implementation of the entire workflow for a more exhaustive validation.

Furthermore, the proposed methodology has been practically tested through its use in a WWTP project, being able to validate it through its use in another project of a different scope would make its application much safer in a real scenario.

References

[1] Vasja Roblek, Maja Meško, and Alojz Krapež. A complex view of industry 4.0. *Sage open*, 6(2):2158244016653987, 2016.

[2] Francisco Almada-Lobo. The industry 4.0 revolution and the future of manufacturing execution systems (mes). *Journal of innovation management*, 3(4):16–21, 2015.

[3] Ana C Pereira and Fernando Romero. A review of the meanings and the implications of the industry 4.0 concept. *Procedia Manufacturing*, 13:1206–1214, 2017.

[4] Bhagya Nathali Silva, Murad Khan, and Kijun Han. Internet of things: A comprehensive review of enabling technologies, architecture, and challenges. *IETE Technical review*, 35(2):205–220, 2018.

[5] Wazir Zada Khan, MH Rehman, Hussein Mohammed Zangoti, Muhammad Khalil Afzal, Nasrullah Armi, and Khaled Salah. Industrial internet of things: Recent advances, enabling technologies and open challenges. *Computers & Electrical Engineering*, 81:106522, 2020.

[6] Hanan Ahmed, AA Ramadan, EH Elkordy, and Ahmed A Elngar. Introduction to industrial internet of things (iiot). In *Industrial Internet of Things*, pages 1–18. CRC Press, 2022.

[7] Sarfraz Nawaz Brohi, Mohsen Marjani, Ibrahim Abaker Targio Hashem, Thulasyammal Ramiah Pillai, Sukhminder Kaur, and Sagaya Sabestinal Amalathas. A data science methodology for internet-of-things. In *International Conference for Emerging Technologies in Computing*, pages 178–186. Springer, 2019.

[8] Chung-Ho Su and Ching-Hsue Cheng. A hybrid fuzzy time series model based on anfis and integrated nonlinear feature selection method for forecasting stock. *Neurocomputing*, 205:264–273, 2016.

[9] Liang-Ying Wei. A hybrid anfis model based on empirical mode decomposition for stock time series forecasting. *Applied Soft Computing*, 42:368–376, 2016.

[10] Daniel Zurita, Miguel Delgado, Jesus A Carino, and Juan A Ortega. Multimodal forecasting methodology applied to industrial process monitoring. *IEEE Transactions on Industrial Informatics*, 14(2):494–503, 2017.

[11] Gökhan Civelekoglu, NO Yigit, E Diamadopoulos, and M Kitis. Modelling of cod removal in a biological wastewater treatment plant using adaptive neuro-fuzzy inference system and artificial neural network. *Water Science and Technology*, 60(6):1475–1487, 2009.

[12] Maneesha V Ramesh, KV Nibi, Anupama Kurup, Renjith Mohan, A Aiswarya, A Arsha, and PR Sarang. Water quality monitoring and waste management using iot. In *2017 IEEE Global Humanitarian Technology Conference (GHTC)*, pages 1–7. IEEE, 2017.

[13] Maged M Hamed, Mona G Khalafallah, and Ezzat A Hassanien. Prediction of wastewater treatment plant performance using artificial neural networks. *Environmental Modelling & Software*, 19(10):919–928, 2004.

[14] Ana Azevedo and Manuel Filipe Santos. Kdd, semma and crisp-dm: a parallel overview. *IADS-DM*, 2008.

[15] Nick Stamatis, Dimitris Parthimos, and Tudor M Griffith. Forecasting chaotic cardiovascular time series with an adaptive slope multilayer perceptron neural network. *IEEE transactions on biomedical engineering*, 46(12):1441–1453, 1999.

[16] Masumi Ishikawa and Teppei Moriyama. Prediction of time series by a structural learning of neural networks. *Fuzzy Sets and Systems*, 82(2):167–176, 1996.

[17] Serkan Aras and İpek Deveci Kocakoç. A new model selection strategy in time series forecasting with artificial neural networks: Ihts. *Neurocomputing*, 174:974–987, 2016.

[18] Zhaozong Meng, Zhipeng Wu, Cahyo Muvianto, and John Gray. A data-oriented m2m messaging mechanism for industrial iot applications. *IEEE Internet of Things Journal*, 4(1):236–246, 2016.

[19] Sotirios Katsikeas, Konstantinos Fysarakis, Andreas Miaoudakis, Amaury Van Bemten, Ioannis Askoxylakis, Ioannis Papaefstathiou, and Anargyros Plemenos. Lightweight & secure industrial iot communications via the mq telemetry transport protocol. In *2017 IEEE Symposium on Computers and Communications (ISCC)*, pages 1193–1200. IEEE, 2017.

[20] Theofanis P Raptis and Andrea Passarella. A distributed data management scheme for industrial iot environments. In *2017 IEEE 13th International Conference on Wireless and Mobile Computing, Networking and Communications (WiMob)*, pages 196–203. IEEE, 2017.

[21] M Carmen Lucas-Estañ, Theofanis P Raptis, Miguel Sepulcre, Andrea Passarella, Cristina Regueiro, and Oscar Lazaro. A software defined hierarchical communication and data management architecture for industry 4.0. In *2018 14th Annual Conference on Wireless On-demand Network Systems and Services (WONS)*, pages 37–44. IEEE, 2018.

[22] Raza Abid Abbasi, Nadeem Javaid, Muhammad Nauman Javid Ghuman, Zahoor Ali Khan, Shujat Ur Rehman, et al. Short term load forecasting using xgboost. In *Workshops of the International Conference on Advanced Information Networking and Applications*, pages 1120–1131. Springer, 2019.

[23] Michael Horrell, Larry Reynolds, and Adam McElhinney. Data science in heavy industry and the internet of things. 2020.

[24] UVBR Thissen, R Van Brakel, AP De Weijer, WJ Melssen, and LMC Buydens. Using support vector machines for time series prediction. *Chemometrics and intelligent laboratory systems*, 69(1-2):35–49, 2003.

[25] Kyoung-jae Kim. Financial time series forecasting using support vector machines. *Neurocomputing*, 55(1-2):307–319, 2003.

[26] Yevgeniy Bodyanskiy and Olena Vynokurova. Hybrid adaptive wavelet-neuro-fuzzy system for chaotic time series identification. *Information Sciences*, 220:170–179, 2013.

[27] Wenwu He, Zhizhong Wang, and Hui Jiang. Model optimizing and feature selecting for support vector regression in time series forecasting. *Neurocomputing*, 72(1-3):600–611, 2008.

[28] Yan Wang and Yuankai Guo. Forecasting method of stock market volatility in time series data based on mixed model of arima and xgboost. *China Communications*, 17(3):205–221, 2020.

[29] Pedro Henrique Borghi, Oleksandr Zakordonets, and João Paulo Teixeira. A covid-19 time series forecasting model based on mlp ann. *Procedia Computer Science*, 181:940–947, 2021.

[30] Yujie Fang, Hui Xu, and Jie Jiang. A survey of time series data visualization research. In *IOP Conference Series: Materials Science and Engineering*, volume 782, page 022013. IOP Publishing, 2020.

[31] Ana Dalia Pano-Azucena, Esteban Tlelo-Cuautle, and Sheldon X-D Tan. Prediction of chaotic time series by using anns, anfis and svms. In *2018 7th International Conference on Modern Circuits and Systems Technologies (MOCAST)*, pages 1–4. IEEE, 2018.

[32] Kenya Andresia De Oliveira, Alvaro Vannucci, and Elton Cesar da Silva. Using artificial neural networks to forecast chaotic time series. *Physica A: Statistical Mechanics and its Applications*, 284(1-4):393–404, 2000.

[33] Fi Takens. Lecture notes in mathematics. *by DA Rand and L.-S. Young Springer, Berlin*, 898:366, 1981.