Smart sensor systems for environmental monitoring: implications and applications

Politecnico di Bari

Department of Electrical and Information Engineering

Electrical and Information Engineering
Ph.D. Program

SSD: ING-INF/05 - Telecommunications

Final Dissertation

Smart sensor systems for environmental monitoring: implications and applications

---

by

Angelo Cardellicchio

Referees

Prof. Giuseppe Pirlo

Dr. Mariofanna Milanova

Supervisors

Prof. Eng. Cataldo Guaragnella

Dr. Tiziana D'Orazio

Coordinator of Ph.D. Program:
Prof. Eng. Alfredo Grieco

Course n. 31, 01/11/2015 - 31/10/2018

Department of Electrical and Information Engineering

Electrical and Information Engineering
Ph.D. Program

SSD: ING-INF/05 - Telecommunications

Final Dissertation

Smart sensor systems for environmental monitoring: implications and applications

---

by

Angelo Cardellicchio

Referees

Prof. Giuseppe Pirlo

Dr. Mariofanna Milanova

Supervisors

Prof. Eng. Cataldo Guaragnella

Dr. Tiziana D'Orazio

Coordinator of Ph.D. Program:
Prof. Eng. Alfredo Grieco

Course n. 31, 01/11/2015 - 31/10/2018

Abstract

Climate change is one of the biggest challenges that humanity will face in the upcoming decades. Hence, over the last few years, the environmental engineering research community has focused its effort on the development and deployment of (often distributed) smart sensor systems, specifically designed for environmental monitoring. These sensors produce large amounts of data, which can be used to describe climate changes and, hopefully, suggest future actions to prevent further damages to the environment. However, to enable the 'smart' capabilities in such systems, researchers must pay attention to several aspects, including two on which this thesis work is focused. The first one, which is often underestimated, is the design of the data acquisition phase: a poor experimental setting will lead to biased data, and therefore ineffective results. The second one concerns the algorithm used to model data, which should be chosen to reflect their intrinsic nature. This work tries to give a first contribution to both these aspect, describing the results of two specific use case scenarios, and highlighting how experiments can greatly benefit from some simple, yet effective, design guidelines. The final goal is to define an initial working pipeline for environmental data processing, which can be both flexible to be adapted to different scenarios, and accurate enough to give an effective description of the observed phenomena.

Acknowledgements

The first, and the biggest, thanks is to my family and girlfriend, Ilaria, for supporting me throughout all the difficulties I have been through these years. Then, a big thank should go to my friends, who still tolerate me despite my lunatic behavior. Without them, this work would not have been possible.

Then, I would like to express my gratitude to my supervisor, Prof. Dino Guaragnella, for his guidance and comprehension, and to my fellow researchers, Giuseppe Dentamaro, Angela Lombardi and the upcoming PhD candidate Matteo Palier, for all the good moments spent together.

Last but not least, a big thanks should go to two special pets, which will always have a place in my heart and soul: Ugo, my dog, and Naja, which is not exactly my dog, but still is a very, very good friend to me.

0.1 Statement of Originality

This is to certify that, to the best of my knowledge, the content of this thesis is my own work. This thesis has not been submitted for any degree or other purposes.

I certify that the intellectual content of this thesis is the product of my own work, and that all the assistance received in preparing this thesis and sources have been acknowledge.

0.2 Publications

Here is a list of publications strictly related to this thesis.

Cardellicchio, A., Dentamaro, G., Di Lecce, V., Guaragnella, C., and Rizzi, M. (2016, June). An opportunistic sensor network approach to wide area environmental sensing. In Environmental, Energy, and Structural Monitoring Systems (EESMS), 2016 IEEE Workshop on (pp. 1-6). IEEE.

Di Lecce, V., Petruzzelli, D., Guaragnella, C., **Cardellicchio, A.**, Dentamaro, G., Quarto, A., ... and Dario, R. Real-time monitoring system for urban wastewater. In Proc. of 2017 IEEE Workshop on Environmental, Energy, and Structural Monitoring Syst.

Cilenti, L., Dario, R., Dentamaro, G., Di Lecce, V., Guaragnella, C., **Cardellicchio, A.**, ... and Specchiulli, A. (2018, March). Sea water distributed monitoring system: A proposal for architecture and data format. In 2018 IEEE International Conference on Environmental Engineering (EE) (pp. 1-7). IEEE.

Blonda, M., Calabrese, A., **Cardellicchio, A.**, Casale, B., Vincenzo, G. D., Lecce, D., ... and Petruzzelli, D. (2018, April). Innovative Methodology for Detecting of Possible Harmful Compounds for Wastewater Treatment the MAUI Project. In 2018 Workshop on Metrology for Industry 4.0 and IoT (pp. 1-6). IEEE.

Cardellicchio, A., Lombardi, A., Guaragnella, C., An Iterative Complex Network Approach for Chemical Gas Sensor Array Characterization. To be published to to Special Section on Industry 4.0: the DIGITAI Transformation in the Engineering findings, IET.

Mali, M., Ungaro, N., Dell'Anna, M.M., Romanazzi, G., **Cardellicchio, A.**, Mastrorilli, P., Damiani, L., Long-term monitoring programs to assess environmental pressures on coastal area: weighted indexes and statistical elaboration, as handy tools for decision-makers. Submitted to Ecological Indicators - Integrating Sciences for Monitoring, Assessment and Management, Elsevier. *Under review.*

Mali, M., Ungaro, N., Dell'Anna, M.M., Romanazzi, G., **Cardellicchio, A.**, Mastrorilli, P., Damiani, L., Supporting materials for Long-term monitoring programs to assess environmental pressures on coastal area: weighted indexes and statistical elaboration, as handy tools for decision-makers. Submitted to Data in Brief, Elsevier. *Under review.*

‘E quindi uscimmo a riveder le stelle.’

Dante Alighieri, Inferno, XXXIV, 139

Contents

List of Tables

List of Figures

Chapter 1

Motivations behind this work

While this thesis is being written, the 'Global Climate Change' informative by NASA [1] states that the global temperature has risen of about 0.45 Celsius degrees since 1880, the Arctic Ice is reducing at a constant rate of 13.2% per decade, and the sea level is increasing by 3.2 millimeters per year.

The trend is clear: humanity, with his way of living, is leading Earth towards a catastrophe. Air and seas are being polluted indiscriminately, radioactive wastes are dumped without any concrete disposal plan, and plastics is let degrade within the oceans. Even if the ozone layer is slowly recovering, there are increasing emissions of carbon dioxide within the air, and future trends do not appear to be promising [2].

Nevertheless, most of the scientific community is seamlessly throwing an alert. Here is a citation by Klein:

> Our economic system and our planetary system are now at war [...]

That means that we can either:

> [...] allow climate disruption to change everything about our world, or change pretty much everything about our economy to avoid the fate.

Concrete action must be taken, to change the economic and social system, protecting environment while keeping current high life standards.

However, to put in place concrete strategies to for environmental monitoring, its current status must be monitored through proper methodologies. This is not an easy task: environment is an incredibly complex system, with a high number of variables, whose interaction causes sudden, unpredictable (and, often, dramatic) changes.

This work summarizes the analysis, experiments and researches conducted to address mainly three aspects of environmental engineering, giving an initial contribution towards a complete set of out-of-the-box tools for environmental monitoring and data analysis.

The first contribution concerns a proposal for a methodology able to deal with the issues related to data transmissions in networks made by distributed environmental sensors. The mathematical foundation of this approach have therefore been defined, and an initial assessment is presented. The second contribution depicts the idea behind the development and deployment of an electronic nose called *Vapor Phase electronic Nose* (*VPeN*), describing its possible application scenarios along with the challenges it tries to address. Finally, the third, and more significant, contribution lies in the initial definition of a *data-driven* approach to environmental data analysis, which can hopefully create a foundation on which enhancements to the current sampling and analysis methodologies will be built. Obviously, the impact of the proposed methodology is limited to the small set of real use cases to which it has been applied; however, this preliminary assessment highlights the need an *extremely compelling need* for a data-driven pipeline for environmental data analysis.

The rest of this work is structured as follows. In chapter 2, an overview on the theoretical background which has led this thesis is described. In chapter 3, a perspective on related works is also given. Then, in chapter 4, the main experimental results achieved during are shown. Finally, in chapter 5, there will be a brief discussion on the implications of this work, and on how it can be further improved in the future.

Chapter 2

Background

This chapter describes the theoretical background on which this work is based.

The first section will introduce a brief overview on the topic of spectrum sensing in the context of opportunistic radio. The, section 2.2 will briefly describe the principles which lead the development of gas sensor arrays.

Afterwards, section 2.3 will introduce the concepts needed for the analysis and interpretation of data acquired by environmental sensors. The discussion will first describe the principle of *Exploratory Data Analysis* (EDA) (section 2.3.1), which has been used as a basis for part of the experimental section of this thesis. Then, a gentle introduction to the specific techniques used for data analysis will be given in section 2.3. Specifically, in section 2.3.4 there will be a focus on the analysis of data when interpreted as independent samples, while time series analysis will be described in section 2.3.5. Finally, the foundations to the mathematical tools called complex networks will be given in section 2.4.

2.1 Spectrum sensing in Environmental Sensor Networks

2.1.1 Environmental Sensor Networks

Environmental Sensor Networks (ESN) [113] are a specific application of the concept of *Wireless Sensor Networks* (WSN) [112] in an environmental use case.

This type of network allows for the acquisition and management of large (*big*) quantities of environmental data, which can then analyzed for several purposes. Let us highlight that an ESN is, by definition, *distributed*: that is, sensors within the network are geographically located apart from each other. The data analysis techniques described in this work can be extended to data acquired by such a network: however, one should consider a proper preprocessing step, to envisage for dynamic conditioning parameters (such as different acquisition settings, clock synchronization, different hardware usage, etc.).

A typical ESN is composed by three layers [11]:

- a *local layer*, made by distributed sensors which gather data and send them towards the next layer;
- an *intermediate layer*, where local concentrators store data sent by the sensors of the local layer;
- a *cloud-based information system layer*, which performs all the necessary analysis.

It is also important to underline that these layers are not tied to a specific technology, even if, in modern architectures, the use of paradigms such as RESTful communication and distributed computing should be more advisable.

However, apart from the specific implementation, the development and deployment of an ESN poses several, heterogeneous, issues. Examples of such challenges are the identification of frequency slots that can be used by the local layer to send data towards the intermediate layer,

or the evaluation of the wireless bandwidth availability needed to allow high speed, short range communication between sensors.

To specifically address these issues, *cognitive* (or *opportunistic*) *radio* approaches have been developed[114].

2.1.2 Cognitive radio

The main aims of cognitive radio are the correct estimation of both the spectrum usage in a given time slot and the bandwidth needed at the receiver end. It is therefore important to specify a set of requirements for such a system; thus, one should first classify the frequency slot under analysis, which can belong to one of three different types [115]:

- *white spaces*, that is, frequency slots where no radio-frequency interferences are found (apart from white noise due to natural and artificial sources);
- *gray spaces*, that is, frequency slots which are partially occupied by radio-frequency interferences or noise;
- *black spaces*, that is, frequency slots which are totally occupied by either radio-frequency interferences or noise.

Another important aspect that cognitive radio approaches must take into account is that the distribution of white, gray and black spaces can vary either in time or space. Finally, it is unlikely that pure white spaces exist in commercially-available bandwidths, due to the wide range of possible interferences.

All these considerations lead to a more realistic scenario, where the following set of non-parametric characteristics is required by a cognitive radio network:

- first, the cognitive radio network should be able to *classify* each spectrum hole, either as a white, gray or black space, with an adequate degree of confidence, which depends by the application;

- second, the *spectral resolution* of the cognitive radio system should be accurate enough to achieve an efficient use of available bandwidth;

- third, the cognitive radio network should be able to estimate the *direction of arrival* of the signals of each interferer, to provide the whole system with information concerning its location;

- finally, the cognitive radio network should be able to exploit cyclostationarity, and use it to reinforce both spectrum hole detection and signal classification, when the band of interest is occupied by a primary user.

2.1.3 Spectrum sensing principles

Spectral estimation first envisages for a two-step preprocessing [11], where the spectrum of the signal is first shifted of an amount equals to the frequency under analysis, and then low-pass filtered, taking a sample at the center of the filtering window for analysis.

This procedure can be carried out through an algorithm which implements the Fast Fourier Transform (FFT), such as the Cooley-Tukey algorithm [3]. However, such algorithm requires a computational load equals to $\frac{N}{2} \cdot \log\left(\frac{N}{2}\right)$, which can considerably grows as N does; therefore, as these algorithms must be implemented at the local layer level, where the hardware often has low power consumption as a requirement (due to the fact that it should be deployed with an embedded battery, which should last as long as possible), an algorithm which guarantees a lower computational load is desirable.

In the approach developed in [11], such issues are addressed by means of a chirp signal. More details will be given in chapter 4.

2.2 Electronic noses for Environmental Monitoring

Like many others in ICT, the concept of *electronic noses* is directly 'borrowed' by nature, and was developed thanks to the efforts made during the 80s while researching machine olfaction.

From a biological perspective, when a nose 'sniffs' a compound, the interaction between odorants and chemical-sensory receptors within the nose triggers several classes of *olfactory neurons* [57]; these, in turn, produce an electrical signal, which is transmitted towards the brain [58]. An interesting consideration is that a single olfactory neuron may respond to several odorants, and each odorant can be sensed by multiple olfactory neurons [59].

These notions lead to the ideation of the electronic nose, which, according to Gardner and Bartlett [60], can be defined as a *[...] measurement system, composed by an array of (gas) sensors, each one of which is (partially) delegated to sense a specific set of compounds [...].* An electronic nose is therefore capable of recognizing both simple and complex odors, thanks to pattern recognition algorithms.

2.2.1 Composition

An electronic nose is usually composed by three systems. The first one is called *sample delivery unit*, and is used to transfer volatile molecules from the source to the sensor array, which is usually fixed within a measurement chamber under constant temperature and humidity conditions. The second one is the *detection unit*, which consists of an electro-chemical transducer, whose outputs is given by a properly filtered electric signal. Then, the third system is the *processing unit*, which embeds a computing unit (e.g. a system-on-a-chip, a micro controller, etc.), whose main role is to process these data, and send them towards an external storage unit.

2.2.2 Operating principles

Sensors generally operate according to the principle which states that *a change within the environments modifies the properties of the sensor in a measurable way.*

Gas sensors obey to this rule, yet, there are several types of principles that they can exploit. In the following, a brief description of each one of these principles will be given[65].

Conductivity-based gas sensors. These sensors exploit the principle that a change in some physical property of the materials used in the sensor leads to a change in its resistance as well. While the mechanisms which lead to these changes are different for each type of material, sensors of this family rely on the same structure and layout.

Specifically, these sensors are composed by a *heater* (often used when the sensor uses metal-oxide materials, due to the high temperatures involved), and a *sensing element*, which is deposited over two electrodes, measuring the relative resistance between them.

Conductivity-based gas sensors use conducting polymers composite as sensing elements. These materials change their resistance due to percolation effects, that is, the vapor permeates the polymer and causes its expansion, with a consequent variation in electric resistance [66]. Another type of polymeric material on which these sensors are based are *intrinsically conducting polymers* (ICP), which operates according to the principle that the absorption of the odorant into the ICP alters its conductivity [67].

Metal-oxide sensors are also used. These are based on variations in the conductance of the oxide when it interacts with a gas; obviously, such variations are proportional to the concentration of the gas itself. Their working principle somehow resemble the one of traditional MOSFET [68].

Polymeric sensors can operate at room temperature, and are cheap; however, it is important that the measurement chamber has a proper thermal isolation. On the other hand, metal-oxide sensors require high working temperature, and suffer from gas poisoning; however, their response are faster than the ones from polymeric sensors.

Mass-based gas sensors. Mass-based gas sensors belong to two different classes: the *surface acoustic wave* (SAW) sensors, and the *quartz crystal microbalance* (QCM) sensors.

The first type of device is composed by an input and an output digital transducer, between

which an acoustic wave with a fixed frequency is sent. A sensitive membrane is placed between the input and the output transducers; when the sensing element interacts with a compatible analyte, it changes its mass, with a consequent change in the frequency of the acoustic wave [69].

QCM are also based on a principle which is similar to SAW; however, in this case, the sensing element is a quartz crystal, which oscillates at its resonant frequency, and whose variations in mass are registered when an interaction with an analyte occurs.

SAW sensors are characterized by high sensitivity and fast responses, but are complex, and measurements are difficult to reproduce. QCM partially address these issues, but are also complex and have a poor SNR.

Optical-based gas sensors. Optical-based gas sensors are based on optical fiber, whose sides are coated with a fluorescent dye, encapsulated in a polymeric matrix. Polarity alterations in the fluorescent dye, or interaction with the vapor, change its optical properties, and can therefore be measured [71]. These sensors are both fast and cheap; however, they are complex, and can suffer from poisoning from photobleaching.

As a final remark, it must be underlined that newly developed sensors are manufactured through micro-fabrication techniques, and are getting increasingly compact, lightweight and inexpensive [61].

In chapter 4, the development and deployment of the VPeN, the electronic nose used in this thesis to analyze a water flow in real time, will be discussed.

2.3 Data Analysis

In this section, the theoretical foundations for the analysis of environmental data will be given.

2.3.1 Exploratory Data Analysis

Data acquired in real-world scenario are not always of easy understanding. In the era of the Big Data, one may easily found him/herself overwhelmed with data, which have only one thing in common: they are *intrinsically complex*. In fact, these data often show noise, redundancies and, in general, their distribution do not resemble a normal one. Therefore, a preliminary analysis of these data is often required: and this is where the concept of *exploratory data analysis* (EDA) comes into help.

The term EDA was coined by John Tukey (who was both a mathematician and a chemist) in 1977 in a work that, since then, has become seminal [4]. Tukey proposed a revolution in the methodological approach to data analysis: instead of relying on *confirmatory* techniques, which had been used until his proposal, EDA suggested the use of a variety of (mostly *visual*) techniques to analyze the characteristics of a dataset.

Confirmatory and exploratory techniques are complementary approaches. On one side, confirmatory techniques are *model-driven*, meaning that they *confirm* that data adhere to a previously established model. On the other side, exploratory techniques are *data-driven*, in the sense that they derive a model *directly from data*.

To understand why EDA has been proposed by Tukey, let us suppose that one must analyze a dataset acquired during a field experiment. Usually, there are protocols to which field experiments must adhere, each one tailored on the specific application. These protocols are obviously created for purposes such as reducing noise within data, allow for experimental reproducibility, and improve the overall quality of the samples. However, following a protocol does not ensure to gather *ideal* data. There are infinite sources of randomness and bias which cannot be under the control of the experimenter. As a consequence, the main assumptions on which most of the models used in confirmatory data analysis, i.e. the normality of the data distribution, *do not (always) hold in real world*.

EDA, however, deals with this randomness by *analyzing the totality* of data, finding patterns and relationships, and creating a meaningful description of the data themselves: as an example,

even outliers are used, as they are representative of *non-experimental* behavior.

It is clear that exploratory and confirmatory data analysis may not be considered as monolithic blocks, but must interact. The role of EDA the exploratory analysis is to extract information from data, and make assumptions, which can be thereafter confirmed by the confirmatory data analysis, whose output can refine the exploratory analysis, and so on, in an iterative procedure.

The application of these techniques will be shown in chapter 4, where they will be used to analyze the results coming from the VPeNs.

2.3.2 Representing and learning from data

Data coming from sensors can be of several different types. However, they can be considered either as *identically and independently distributed*, when the sampling process does not have a (relatively) high sampling rate, or as *time-dependent series*, if the process has a sampling rate high enough to cause dependencies between consecutive samples. Clearly, the knowledge of both the settings and the type of process which is involved in the measurement can help to define the mathematical tools which are most well suited for data analysis. In the following, the process behind this choice will be described.

2.3.3 The importance of the dataset

Every data analysis technique relies on two factors: the *algorithm* and the *dataset*. Choosing the most appropriate algorithm is not trivial, and the motivation behind the choice of each method will be discussed from case to case in chapter 4. However, here there will be a brief focus on why the dataset is also important. To this end, let us describe the experience from the field of computer vision, which has been used as a basis for several concepts in this thesis, and lead some complementary works.

ImageNet [12] has been developed by Fei-Fei Li and her team in the time span between 2007 and 2009. The main goal was to create a dataset which could 'entirely' model the real world,

so a total of more than 13 millions of images, which represented more than 1000 classes of objects, were gathered. In 2012, Alex Krizhevsky introduced AlexNet [13], the first deep convolutional neural network which could achieve more than the 90% of accuracy on ImageNet. From then, deep learning became the 'next big thing', and has been used for nearly every possible application.

However, it was not a novel machine learning algorithm which led towards this success. Convolutional neural networks are indeed a relatively old concept: their introduction can be traced to 1968, when Hubel and Wiesel released a study on the receptive fields associated to each neuron within the monkey striate cortex [14]. Even if AlexNet 'legitimized' several important solutions, which since then had become the standard, such as the use of rectified linear unit for the activation, the (huge) step forward was made by the combination of three factors, i.e. the use of a (sufficiently) deep architecture, the availability of enough computational power to train such a network (thanks to GPGPU), and the dataset itself, which was big enough to ensure that all the parameters of the network could be properly trained.

This leads towards a conclusion: *data are important*. Data science envisage for the knowledge of data, through techniques such as EDA. And the selection of the algorithm can be only *consequential* to an accurate knowledge of the dataset. The implication of this in the framework of the data used in this work will be clear in chapter 4.

2.3.4 Identically and independently distributed data

One of the main assumption for data analysis is that data are identically and independently distributed *iid* [5]. This means that each sample is independent from the others, and must be considered as a single representation of a physical phenomena.

Starting from this, it is crucial to define the difference between several type of analysis which can be carried on iid data. First, there is the difference between *classification* and *regression*:

- *classification* envisage for data belonging to (at least) one *class*. A class is a sort of

'prototype' for data, which are supposed to adhere to the physical characteristics of the class itself;

- *regression*, on the other hand, allows to characterize an algebraic relationship between data.

In this work, only classification techniques will be used. However, between these, it is possible to find further differences, specifically between *supervised* [7] and *unsupervised* [8] learning techniques. In the following, the simple (yet deep) difference between these two types of classification will be described.

Supervised learning

In supervised learning, the algorithm learns a function that maps an input to an output, according to a set of example input-output pairs.

This is a generic definition, which should be properly analyzed. The first, relevant concept is related to the fact that *supervised learning algorithm are learning a function.* Therefore, one could expect that a classifier C learns a (possibly non-linear) mathematical function f which *maps an input x to an output y.* To do that, the classifier C uses an adequate number of input-output pairs, therefore evaluating how x must be 'transformed' by f to be turned into y. This can be put in a simpler, yet elegant, form:

$$C = f : x \to y \tag{2.1}$$

Let us analyze the *form* of both x and y. Supervised learning is often used for classification, where a discrete label $y \in \{1, \dots k\}$ is associated to an n-dimensional vector $x \in R^n$. Therefore, equation 2.1 can be also written following the definition given by Szegedy in [9]:

$$C = f : R^n \to \{1, \dots, k\} \tag{2.2}$$

The above equations can be easily understood through a simple real-case example. Let us suppose that X is the set of samples taken during an experiment with an electronic nose, and that each sample $x_i \in X$ is associated to a label $y_i \in K$. Furthermore, let us suppose that each label y_i describes the substance, known beforehand, to which the sample x_i is referred to. Therefore, equation 2.2 simply states that classification associates a substance to each sample.

Despite the underlying conceptual simplicity, supervised learning algorithms require a rigorous training strategy to be effective. As an example, one must consider that the algorithm tends to adhere to the set of input-output pairs on which is trained: that is, C founds a function f which is fine-tuned to minimize the relative distance between the output of the algorithm $\hat{y}$ and the real labels y. As a consequence, if training data do not adequately represent a wide range of possible cases, the algorithm will suffer from a lack of generalization. Going back to the previous example, if the largest part of X represents a single solution s, while the other part represents all the other substances, C will probably classify samples which refer to s with a high degree of accuracy, while samples referred to other solutions will be plausibly mixed up.

This issue, along with several other related problems, can be addressed making an effort to gather more (and, possibly, more significant) data is needed; specifically, C should be trained on a large number of samples taken by all the possible solutions. Furthermore, a testing procedure is needed to evaluate the generalization capabilities of C. Specifically, testing can be performed either splitting the dataset in two groups (that is, *training data*, on which training is actually performed, and *testing data*, against which the accuracy of the function f is tested), or using a k-fold cross-validation, which is an iterative procedure where, at each iteration, samples are randomly excluded from the training procedure, and, at the end, only the mean value of accuracy is considered.

To classify data used in this thesis work, a deep supervised learning algorithm, based on the concept of artificial neural networks and multi layer perceptron, has been used. The procedure is extensively described in chapter 4.

Unsupervised learning

Unsupervised learning differs from supervised learning mainly as *no labels are given*. As a consequence, it is not possible to evaluate the accuracy of such algorithms simply relying on how much the outputs of f adhere to the ground truth given by the actual labels.

There are several examples of unsupervised learning algorithms. As an example, there are deep neural networks which are based on unsupervised learning, such as autoencoders [120] and Self-Organizing Maps [121].

Another class of widespread unsupervised algorithms is given by *clustering* algorithms. Clustering can be intuitively defined as the task of grouping a set of data objects in such a way that similar objects are grouped together, while different objects are 'pushed away' from each other. This similarity is determined from the *clustering criteria*, which is a metric that should be minimized for objects that belong to the same cluster, while being maximized for objects which belong to different clusters.

Over time, several clustering methods have been developed, each one suited for a specific situation, with a certain type of data. As an example, centroid-based methods, cluster data with respect to the distance between each data point and the centroid of the cluster itself, and are often more suited when clusters are elliptical and normally distributed; on the other side, connectivity models evaluate the connectivity (also known as *linkage*) within each cluster, according to a certain distance metric [122].

In this work, a clustering algorithm has been used to evaluate the goodness of the data acquired by the VPeNs. The whole procedure will be described in chapter 4.

2.3.5 Time series analysis

When the sampling rate of data acquired during an experiment is high enough, samples are no longer independent and identically distributed, but may present some relationships due to

previous effects. Therefore, traditional learning techniques cannot be used, and the concept of *time series* has to be introduced.

A time series can be though of as a series of data point which are listed in time order, with a high sampling rate; another way to represent a time series is as a sequence taken at consecutive points over time. An example of time series is given by spoken language: in a sentence, consecutive words are dependent and, to predict the next words, one should know the context, as given by previous words.

It is important to underline that, usually, time series are assumed to be *evenly spaced*, that is, samples are taken at regular intervals. When time series are unevenly spaced, one can either re-sample the time series to fill missing values [107, 108], use a maximum likelihood function to estimate them [123], or adopt a type of model which specifically deals with this situation, such as GARCH [124]. In this work, as shown in chapter 4, the first approach has been used, mainly as it is the most adopted when dealing with natural phenomena.

Let us now introduce the concept of *time series modeling*, as depicted in [17].

Time series modeling

A time series can be formally defined as a sequence of observation $y_1, \ldots, y_T$ [17], which are assumed to be relative to an unobservable generating process. The modeling of a time series make a fundamental assumption: this generating process is a *stationary stochastic process*.

Let us recall the definition of *stationarity*. A stochastic process $\{y_t\}_{t=1}^{T}$ is defined as *strictly stationary* if the distribution of $\{y_{t+s}\}_{t=1}^{T}$ for an arbitrary value of s has the exact characteristics of $\{y_t\}_{t=1}^{T}$. In other words, a time series is strictly stationary if its distribution is independent from time; obviously, this assumption does not (always) hold in real world.

A more realistic assumption for a real process is *covariance stationarity*. A process $\{y_t\}_{t=1}^{T}$ is covariance stationary if:

- its mean and variance are finite, and independent of time t;

- the autocovariance between time instants t and $t - s$ is finite, and depends only on the lag $\tau = t - s$.

Formally:

$$\begin{aligned} E[y_t] &= \mu < \infty && \forall t \\ V[y_t] &= \sigma^2 < \infty && \forall t \\ \gamma(t,s) &= \gamma[\tau] < \infty && \forall (t,\tau) \end{aligned}$$

Let us focus on the concept of *autocovariance*, which describes the covariance between two values of the stochastic process at different time instants[18]:

$$\gamma(t,s) = Cov(y_t, y_s) = E[(y_t - \mu_t)(y_s - \mu_s)] \tag{2.3}$$

The *autocorrelation function* is given by:

$$\rho(t,s) = \frac{\gamma(t,s)}{\sigma_t \sigma_s} \tag{2.4}$$

The value of $\rho(t,s)$ can span from -1 to 1, and is a better measurement for the dependency structure, as autocovariance is strongly related to the actual values of the signal and, in some cases, does not necessarily give an indication about the relationship between observation in different time instants.

Visualizing autocorrelation. Let us give a simple example to better explain how autocorrelation can be used to evaluate stationarity through visual techniques.

First, let us start by considering figure 2.1, which shows the time representation of the Passengers Dataset used by Box and Jenkins [19].

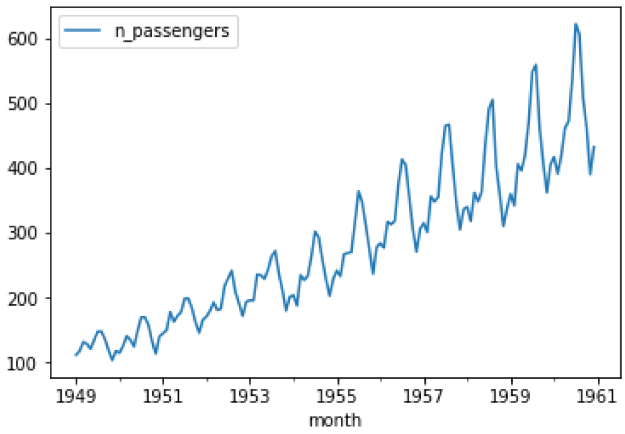
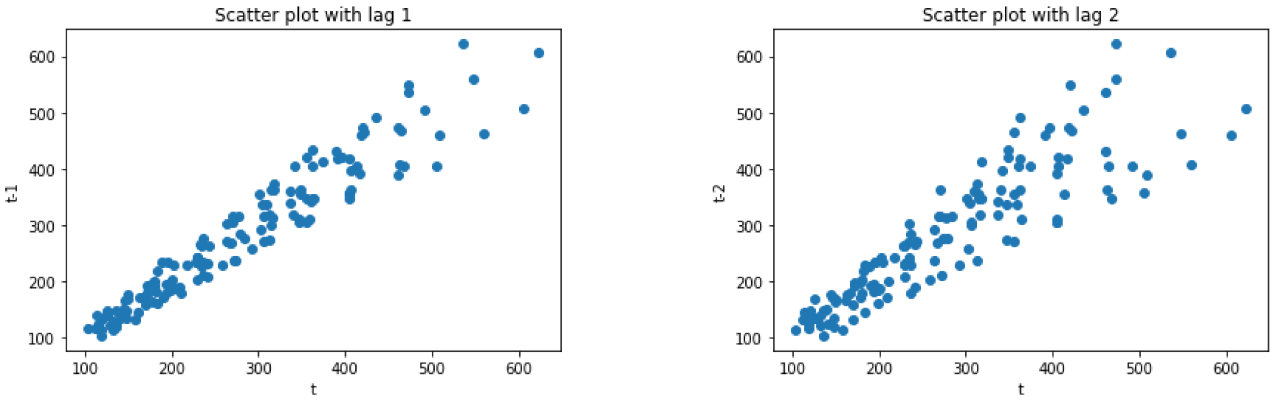


Figure 2.1: The time representation of the Passengers Dataset from Box and Jenkins.

At a first glance, figure 2.1 appear to be non-stationary, due to the values which constantly increase over time. However, let us consider figures2.2a and 2.2b, which represent the scatter plots of the number of the passengers at time t against time $t-1$ and time $t-2$, respectively.



(a) Lag $s = 1$

(b) Lag $s = 2$

Figure 2.2: Scatter plots of the Passenger Datasets

There are strong evidences that there are linear dependencies for the two lagged version of the time series. If one continues to check scatter plots, linear dependencies will be found until at least $s = 9$ [19]. This leads to a conclusions: autocorrelation effects are, sometimes, not evident through a simple analysis.

Obviously, it is unrealistic to assume that one can draw the scatter plot until non-linear dependencies between lagged versions of the time series show up. Therefore, an useful set of tools is given by the *autocorrelation function* (ACF) and the *partial autocorrelation function* (PACF) [19].

The ACF is a function of the time displacement of the time series itself. Informally, it represents the similarity between lagged observation as a function of the time lag s between them. The PACF, instead, represents the conditional correlation between two variables, under the assumptions that the effects of all previous lags on the time series are known. As it will be shown in the following, ACF and PACF can be used to define to which class of process the time series under analysis belongs.

Types of time series processes

White noise processes. The first type of process is *white noise*. In a white noise process, each time sample has a probability distribution with zero mean and finite variance.

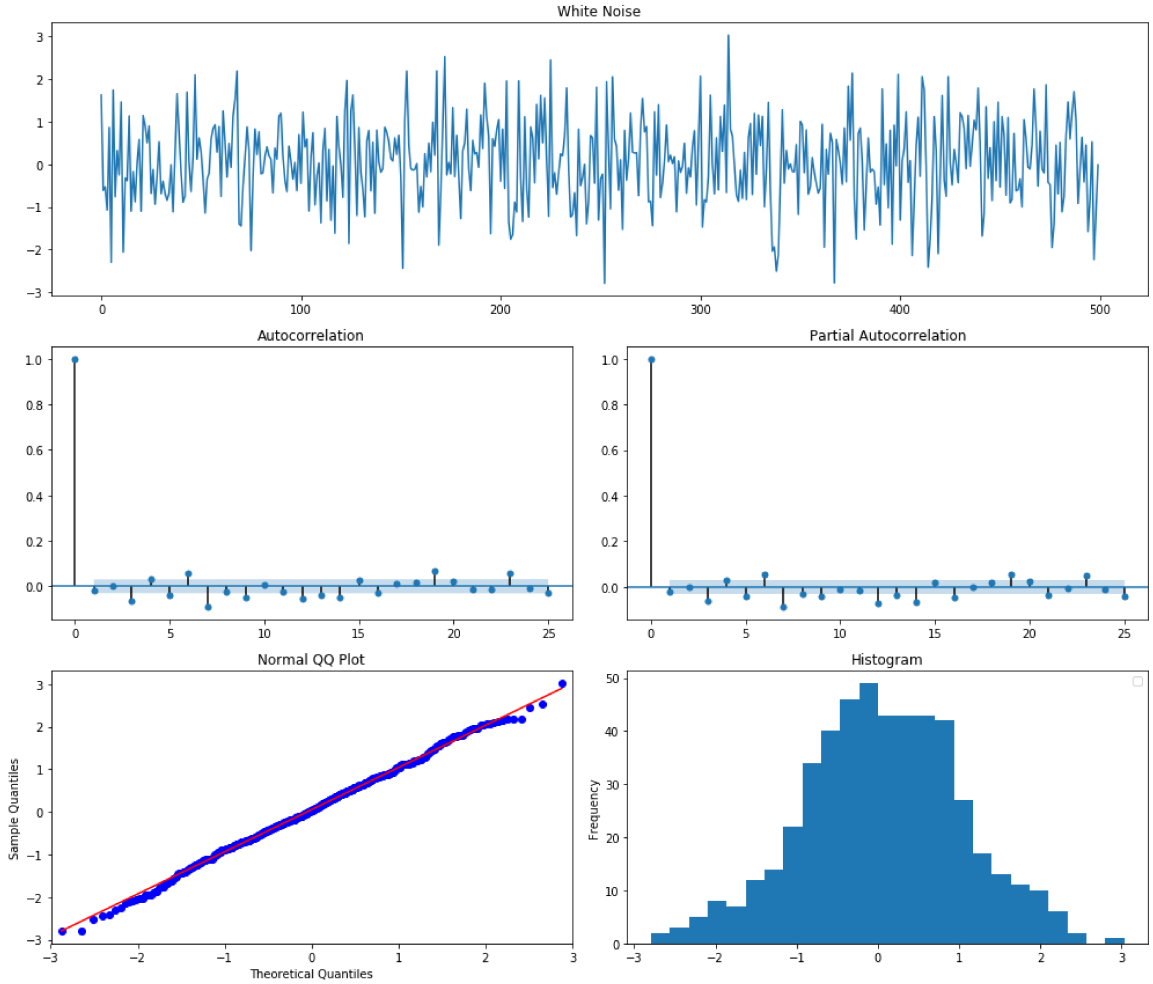


Figure 2.3: The EDA for a random process.

Furthermore, time samples are uncorrelated.Such processes can be expressed as $x_t = \varepsilon_t$, where

$\varepsilon_t \sim (0, \sigma^2)$. Figure 2.3 shows an example of a white process. It can be seen that both the ACF and the PACF of white noise processes only have a spike at $s = 0$, meaning that the process at time x_t is uncorrelated with the value of the process at any other time lag. Intuitively, the QQ plot and the histogram indicate a normally-distributed behaviour.

Random walk. The second type of process is called *random walk*. The expression for a random walk is $x_t = x_{t-1} + \varepsilon_t$, meaning that consecutive time samples are correlated by white noise. This type of process is completely non-stationary, and, as such, time series governed by random walks are unpredictable. Random walks usually show a slowly decreasing ACF, while PACF dramatically drops after the first lag. It is also important to underline how both the normal Q-Q plot and the histogram clearly show that the process is not normally distributed.

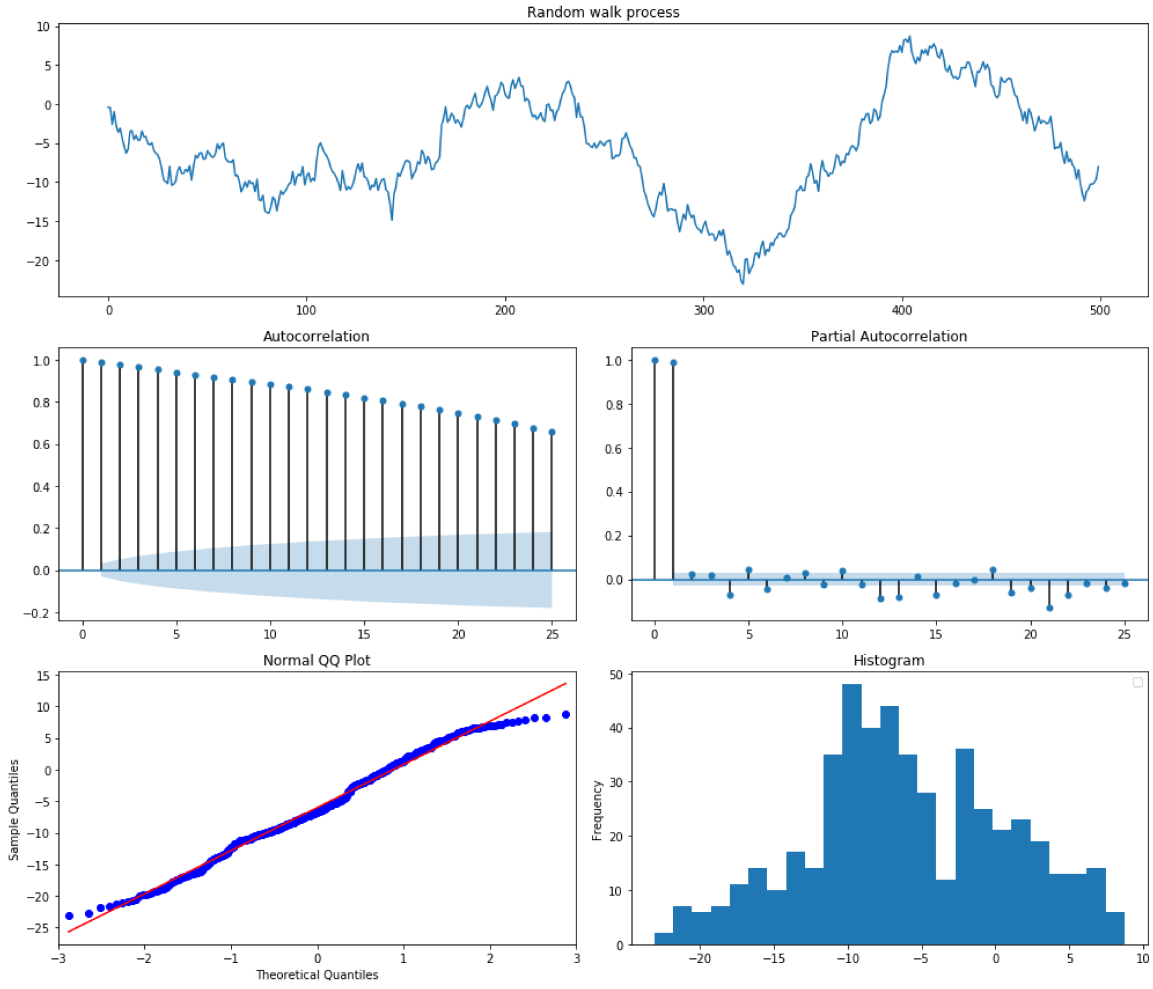


Figure 2.4: The EDA for a random walk.

Autoregressive process. Random walks belongs to a more general group of processes, called *autoregressive* (AR) processes, which have the following form:

$$x_t = \alpha + \sum_{i=1}^{p} a_p x_{t-p} + \varepsilon_t \tag{2.5}$$

Equation 2.5 shows that the value of the process at time t is a linear combination of previous observation, plus a bias term α and white Gaussian noise. In figure 2.5, the EDA for an AR(1) process is shown. As it can be seen, both the ACF and the PACF drops after the first lag. Furthermore, values are normally distributed, due to the presence of the white noise term.

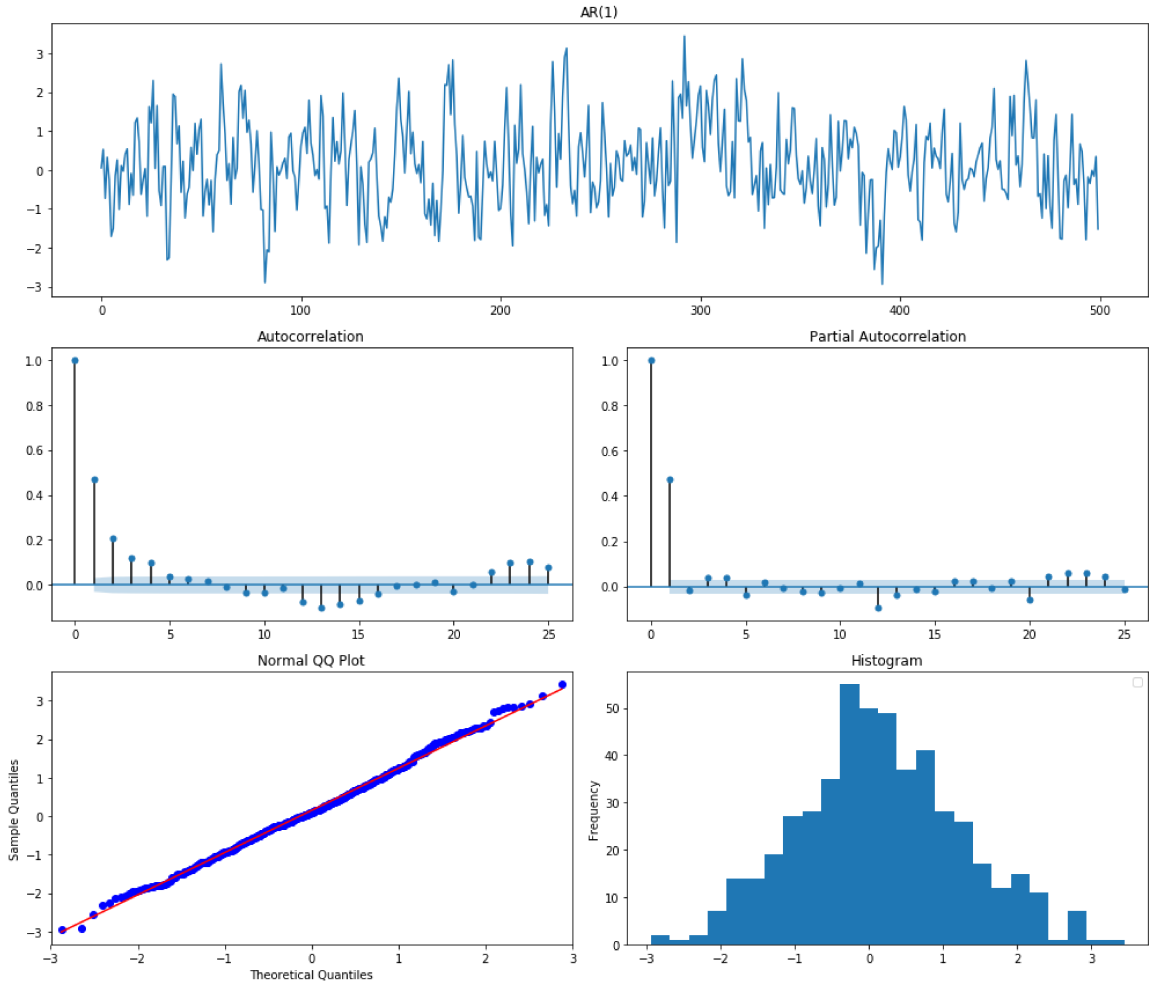


Figure 2.5: The EDA for an AR(1) process.

It can be noted that a random walk is an AR process where $\alpha = 1$. This particular situation is called *unit root*, and it is directly related to non-stationarity.

Moving Average process. A *moving average* (MA) process assumes that the observed time series can be represented by a linear combination of white noise terms:

$$x_t = \varepsilon_t + \sum_{i=1}^{q} b_i \varepsilon_{t-i} \tag{2.6}$$

A MA process is always stationary. In figure 2.6, an MA(1) process is shown. It can be seen that both ACF and PACF quickly decay after the first lag, with a small 'sinusoidal tale' on the PACF. Obviously, both QQ-plot and histogram fit a normal distribution, as the MA process is a linear combination of normally distributed random samples.

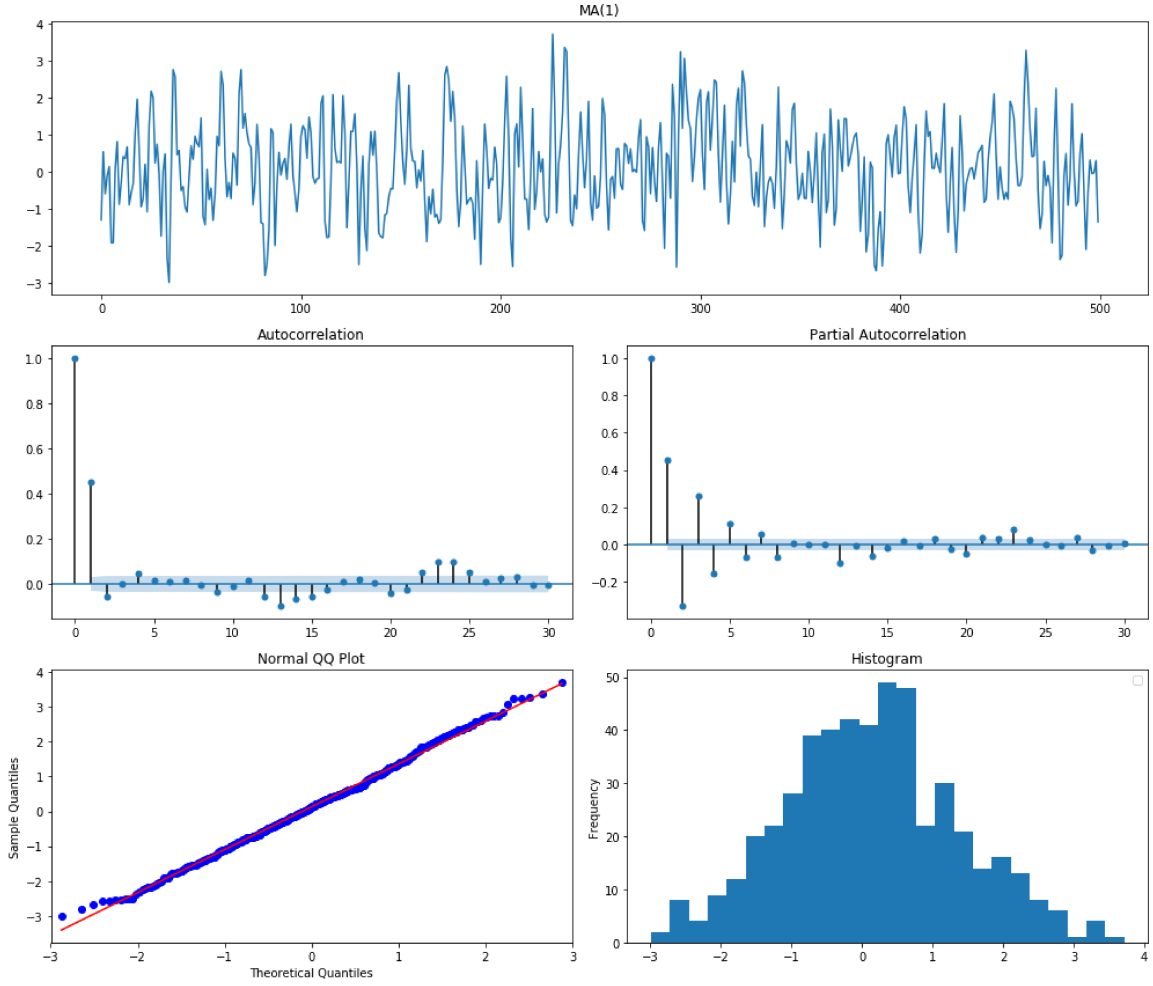


Figure 2.6: The EDA for a MA(1) process.

Let us now perform an EDA for the Passengers Dataset (cfr. figure 2.7. Clearly, the dataset is non-stationary, as it can be seen from the time plot. Furthermore, it resembles a random walk process, and, as a consequence, forecasting cannot be made.

However, let us recall something we mentioned above, that is, random walk processes are autoregressive with a bias term equals to 1. Therefore, there may be a method to transform this process, and allow forecasts. To this end, let us first introduce the concept of *time series decomposition*.

Time Series decomposition

Time series can be decomposed into several components. One of the most used techniques for decomposition is *Seasonal-Trend Decomposition* (STL) [20], which envisage for the following components:

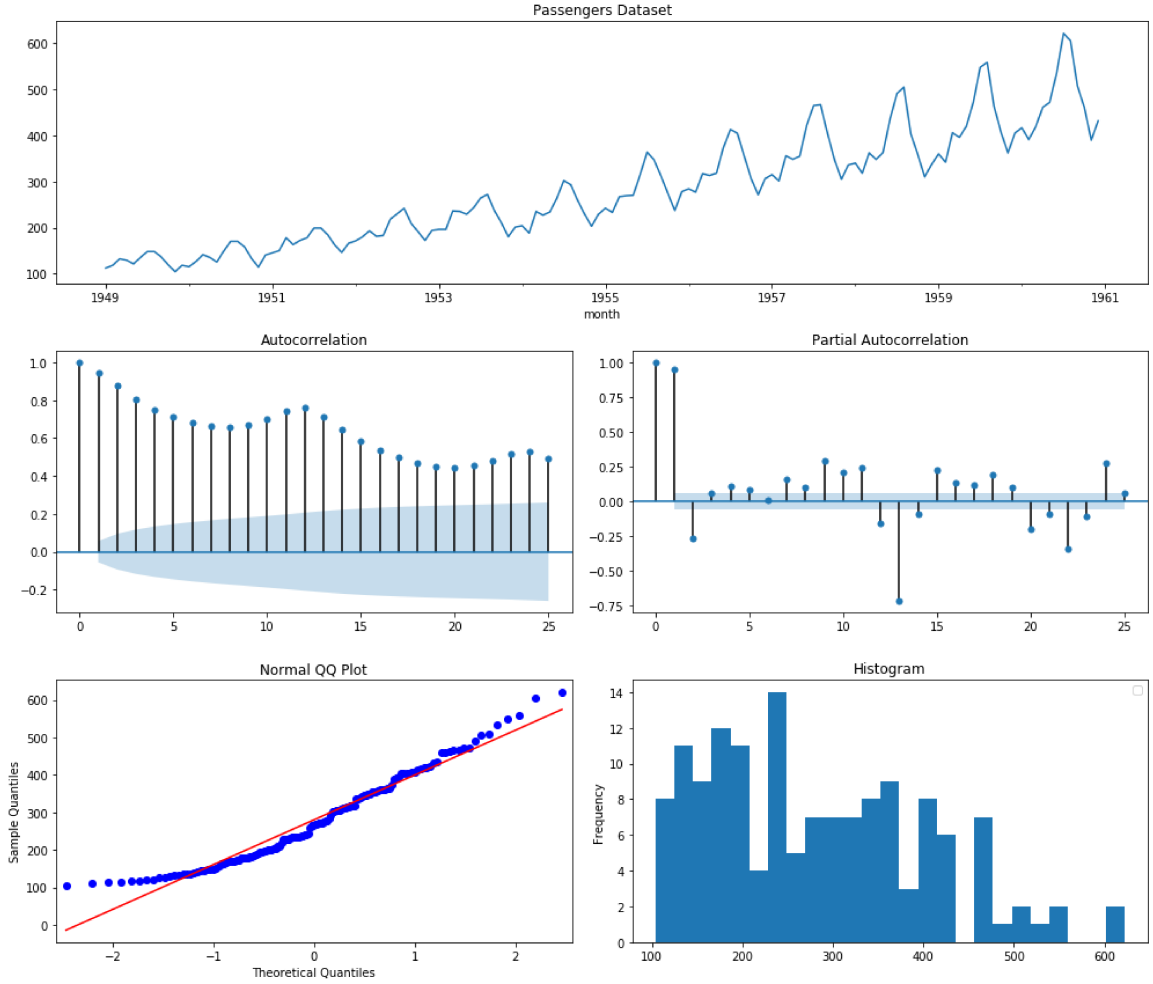


Figure 2.7: The EDA for the Passengers Dataset.

- **trend**, which represents the overall 'direction' of the series;

- **seasonality**, which represent monthly or yearly patterns;
- **noise**, which is an irregular residual left after the extraction of all the components.

There may be another component, referred to as *cycle*, which represents long-term cycles, and is usually found in financial time series [125].

STL decomposition considers either a *multiplicative* or an *additive* composition of three terms, that is, the trend, the seasonal component, and the residuals. The difference between multiplicative and additive decomposition is intuitive: in the multiplicative decomposition, the time series is given by $y_t = t_t \cdot s_t \cdot \varepsilon_t$, while in the additive decomposition the time series is given by $y_t = t_t + s_t + \varepsilon_t$.

An example of STL additive decomposition on the Passenger Dataset is shown in figure2.8.

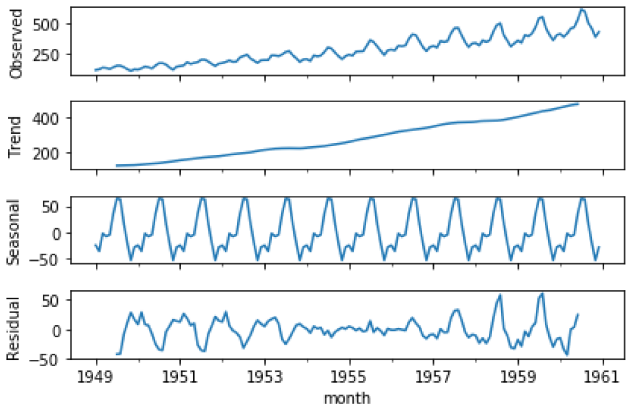


Figure 2.8: Results of an additive STL decomposition for the Passengers Dataset.

It is important to note that STL decomposition is often used only as a visual tool, and more rigorous approaches are available to test for stationarity.

One of the tests most commonly used is the *augmented Dickey-Fuller* (ADF) test[21], whose main purpose is to test the null hypothesis that a unit root is present in a time series data. Let us recall that a time series has a unit root when the characteristic equation has at least one root whose value is 1 [21].

The ADF test tests the following regression model:

$$\Delta y_t = \alpha + \gamma y_{t-1} + \sum_{i=1}^{p} \Delta\beta_i y_{t-i} + \varepsilon_t \tag{2.7}$$

with the hypothesis:

$$H_0 : \gamma = 0$$

$$H_1 : \gamma < 0$$

The results are compared against the Dickey-Fuller test statistics, and if the test statistics is smaller than the critical value (which is usually 5%), the hypothesis is rejected. As an example, running the ADF test against the Passenger Dataset, one obtains a value for the test statistic of 0.81, with a p-value of 0.99, both of which are above the 5% threshold. Therefore, the null hypothesis cannot be rejected, and the series is non-stationary.

Hence, a method to transform a non-stationary series is needed. There are several ways to achieve this: two of the most popular methods consist in the application of a reversible transformation to the the time series (such as a logarithmic transform), and differencing the time series, removing trends from data.

In this work, the latter approach has been used. Specifically, differencing a time series allows to obtain a new time series, by subtracting the value at instant y_{t-k} to the value at instant y_t, with k being the order of the difference. As an example, applying a first-order difference (with $k = 1$) simply means subtracting the value of the time series at the instant t to the value of the time series at the immediately precedent instant. If such transform is applied to the Passenger Dataset, the ADF test on the transformed series will give a value for the test statistics of about -2.83, with a p-value slightly above 0.05, clearly improving further analysis.

Let us evaluate the results of differencing and log-transforming the Passenger Dataset, which are shown in figure 2.9. As it can be seen from ACF and PACF, there are effects which are characteristic of both an AR and a MA process, which should therefore be characterized. This

is possible thanks to *autoregressive-integrated-moving average* (ARIMA) models, which will be described in the following.

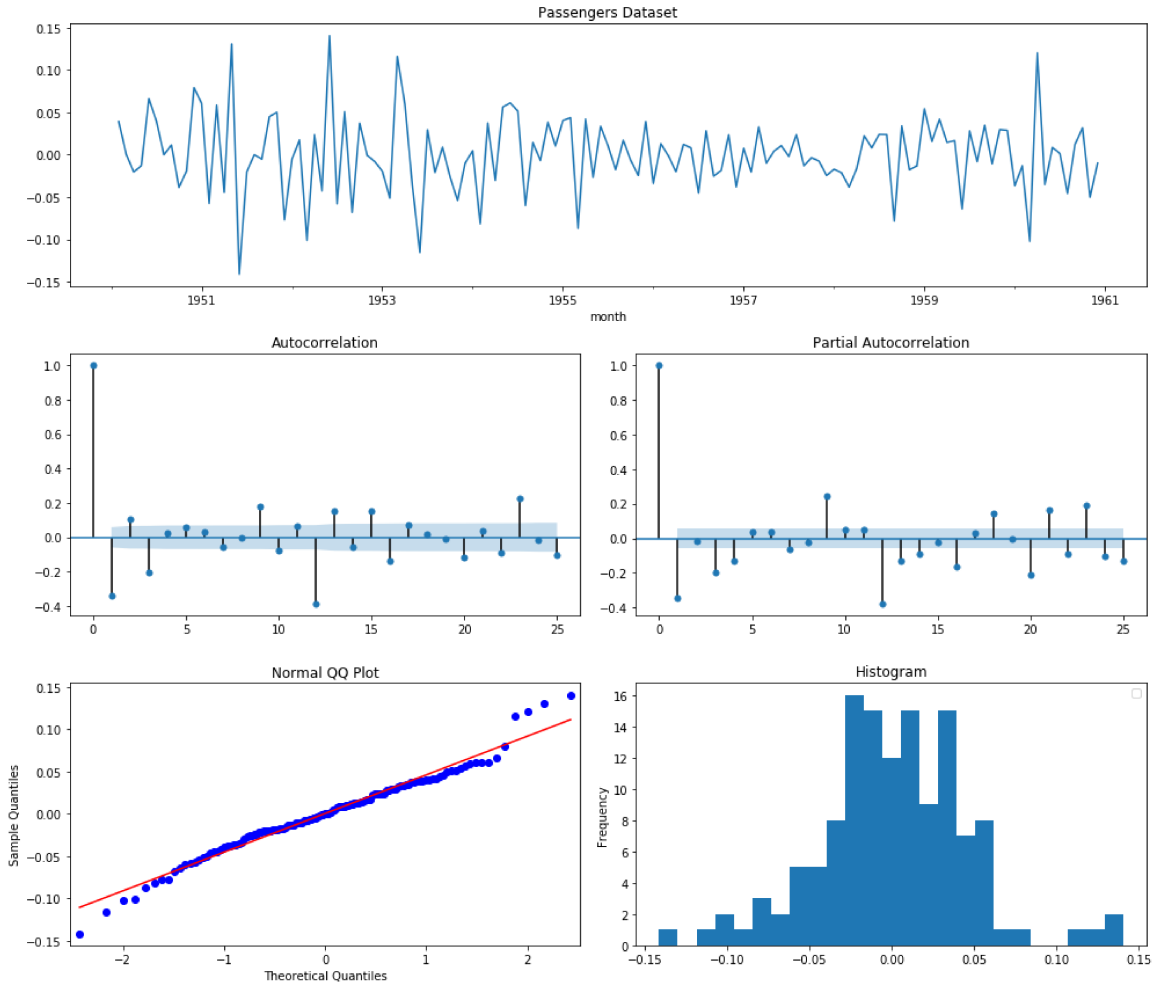


Figure 2.9: EDA for the differenced and transformed Passenger Datasets.

ARIMA modeling and forecasting

ARIMA models[19] are a family of models which are used to model a time series. ARIMA modeling allows to describe all the behaviors described in previous sections, ranging from autoregressive to moving average; furthermore, it allows to automatically transform a non-stationary time series by differencing it.

To achieve this goal, ARIMA models make use of three parameters, namely (p, d, q), which are also known as *orders* of the model. Specifically, p is the order of the AR component component, d is the order of the differencing component (that is, the number of differentiations which should

be considered), while q is the order of the moving average component.

An ARIMA model is described by the following characteristic equation:

$$\left(1 - \sum_{i=1}^{p} \phi_i L^i\right) (1 - L)^d y_t = \mu + \left(1 + \sum_{i=1}^{q} \psi_i L^i\right) \varepsilon_t \tag{2.8}$$

In equation 2.8, the *lagged operator* L performs a lagged transformation of a certain order k, that is:

$$y_t L^k = y_{t-k} \tag{2.9}$$

Equation 2.9 simply states that the lagged operator 'delays' the time series y_t of k time lags.

ARIMA models can be extended to cope with seasonality [19]. These models, which are often referred to as SARIMA models, have four additional parameters, that is, $(P, D, Q)s$, which account for seasonal effects. The meaning of (P, D, Q) is, intuitively, directly related to the orders of the seasonal component of the SARIMA model; as for the s term, it represents the value for seasonality, that is, the number of lags after which a seasonal effect is expected to repeat itself. As an example, if the time series under analysis is sampled on a per-month basis, seasonality should be set to 12, as seasonal effects are expected to be come back after 12 months.

Intuitively, the effectiveness of ARIMA models relies on the correct choice of the parameters. In simple cases, one can effectively choose p and q from the ACF and PACF plots, respectively, while d can be determined by evaluating the temporal plot of the time series. However, in this work, a more complex (yet complete) procedure has been used; it will be described in chapter 4.

2.4 Representing real world systems with complex networks

Real world is filled by examples of complex systems. Most of them are part of daily lives: as an example, social networks, where users interacts in complex, non-linear ways through posts, relationships, file sharing, and much more, are complex systems. The Internet itself, where a huge number of hubs exchange information through packets and streams of data, is a complex system. Nature, also, tends to organize itself in complex systems, such as schools of fishes or storms of birds, where several individuals act as a whole, without an apparent leader, by means of a complex system of interactions. Also human brain is a complex system, whose interactions can be characterized in terms of electrical signals, and have long-term, complex consequences on the way human body operates.

Intuitively, heterogeneous principles which guide each one of the aforementioned systems. However, these all have something in common, as they are ruled by *short-range interactions between components*, whether these are individual, computers, or anatomical parts, and which are non-linearly related to the overall behavior of the system.

It is therefore possible to identify a small set of common properties for complex systems:

- *complex systems are composed by several interacting parts*: a school of fish is made up by several individual fishes;
- *each part in a complex system has its own internal structure*: each fish is an independent organism;
- *the individual behavior of each component affects the whole system in a non-linear way*: a movement of a fish, which spots a predator, can influence the movement of another fish in another part of the school;
- *the relationships between individuals determine the overall behavior of the system*: the movement of the school of fish is directly related to the movement of each fish within it.

It is interesting that all of these properties can be perfectly characterized using just a single mathematical tool: *graphs*.

2.4.1 Complex networks and complex systems

Graphs and *graph theory* are relatively new fields in mathematics. They have been originally developed by Euler, who found a formal solution to the problem of the seven bridges of Knigsberg [15] introducing the concept of graph.

Formally, a graph is a pair $G = (V, E)$, where $V = v_1, v_2, \ldots, v_N$ is a set of N nodes, interconnected by M edges $E = e_1, e_2, \ldots, E_M$. A graph is *weighted* if there is a set of M weights $W = w_1, w_, \ldots, w_M$, each one associated to a specific edge; otherwise, the graph is *unweighted*. Intuitively, weights model the strength of the relationship between nodes: the higher the value for the weight, the greater the strength of the relationship. In unweighted graphs, relationships are therefore supposed to be binary (that is, either the relationship exists or do not exists). Another important distinction is between *directed* and *undirected* graphs. In the first, there is a *direction* associated to each edge e_j, that is, the relationship goes from v_i to v_j. Obviously, this does not hold for undirected graphs.

These concepts have helped to build the theory behind the modeling of complex systems, and have been used to develop the mathematical tools known as *complex networks* [16].

Let us start with a simple example taken directly from the aforementioned complex systems. Social networks can be modeled by a graph, where each node is associated with an user, while an edge between two users states whether they have a friendship relation or not. Clearly, this is an *undirected* and *unweighted* graph, as a friendship is (hopefully) a mutual relationship, and, in its simplest form, is not weighted. Let us consider, however, different types of friendships, such as 'co-worker', 'friend', or 'family member'. Obviously, each one of these types has a different 'strength', and, therefore, this allows to rephrase the graph as a weighted one.

The correspondence between a complex network and a graph is clear: a complex network *is* a graph, and therefore all the algorithms and concepts which are used in graph theory can be

used to model complex networks. In the following, some of the most important concepts will be described.

2.4.2 Properties of complex networks

Complex networks allow to model systems which exhibit chaotic and highly non-linear behaviors; therefore, the early focus on such systems was on the characterization of the spread of information.

In that sense, the first noticeable effort was made in 1958 by Erds Rnyi, who introduced the *Erds-Rnyi* (*ER*) model [29] as a way to generate random graphs. Such structures, an example of which is shown in figure 2.10, are characterized by the property that the probability of having an edge which connects two nodes is the same for all possible pairs of nodes.

To understand the implication of this, let us briefly introduce two concepts, directly inherited from graph theory, that is, *degree* and *degree distribution*. Specifically, the *degree* k of a node v_i is given by the number of nodes which are adjacent to the node itself; the *degree distribution* $P(k)$ is the probability distribution of values k_i for all the nodes in the network.

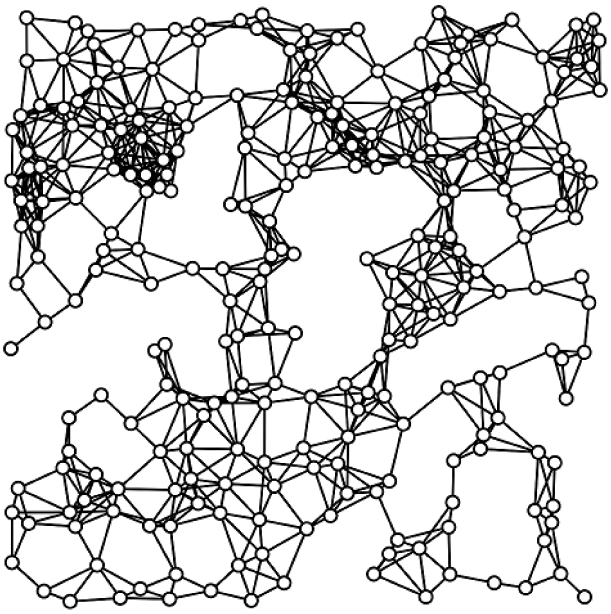


Figure 2.10: A random graph.

Given these definitions, a perfect ER random graph is characterized by an uniform degree distribution. And, obviously, this is not a property which can be found in real complex systems: as an example, it is unlikely that each pair of users in a social network has the same number of relationships.

World is intrinsically heterogeneous, and complex systems often reveals high levels of hierarchical organization. Thus, these system can be characterized by networks with a degree distribution which follows a power law in the following form:

$$P(k) = k^{-\gamma} \tag{2.10}$$

Networks characterized by a degree distribution as in equation 2.10 are called *scale-free networks*, and were discovered by De Solla Price during the studies of citations between scientific papers. However, they were described only some years later, by Barabsi et Bonabeau [30] during a study concerning the World Wide Web.

Scale-free networks are characterized by an interesting property: there are a small number of high-degree nodes, called *hubs*, which connect areas whose nodes are of lower degree. This property has a natural reflection in real world: as an example, in social networks there are few highly-connected hubs, which (indirectly) connect different 'communities' of non-densely connected users. This has also led to the definition of the *small-world* phenomenon, which is well described by the concept of *six degrees of separation*: even if each user has only a small set of connections, the length of the path needed to reach every other user in the network is given by a limited number of steps (usually six).

In this work, the application of complex networks to the modeling of an electronic nose has been described. Unfortunately, data acquired by VPeNs were unsuitable to perform this task; therefore, as it will be described in chapter 4, another affine, publicly available dataset has been used to perform this task.

Chapter 3

Related Works

In this chapter, a perspective on related works is given.

First, in section 3.1, the usage of sensor arrays for environmental monitoring, along with some of the challenges which must be addressed, are described. Then, in section 3.2, techniques which are most commonly used to analyze data coming from gas sensors are shown, while in section 3.3 a particular focus on the analysis of wastewater is given.

3.1 Sensors arrays for environmental monitoring

Sensors, possibly arranged as in a sensor network, have been subject to several studies in the environmental engineering field [133]. In chapter 2, a specific type of sensor array (electronic noses) has been introduced. However, it is important to underline that another type of sensor arrays, called *electronic tongues* [134] is commonly used for environmental monitoring.

The main difference between these two types of sensor arrays lies in the phase in which they operate [102], as electronic noses detect the analyte in gas or vapor phase, whereas electronic tongues work in the liquid phase. Hence, electronic tongues tend to suffer more from poisoning, as they are directly in contact with the analytical sample, while for electronic nose there is a physical separation between the analyte and the sensing element [135]. This is especially

important in the context of wastewater monitoring, as the complex chemical composition of urban and industrial wastewater discourages a direct contact between the sensing elements and the water matrix under analysis [136]; however, using a gas sensor implies that the system should envisage for a mechanism to transform the water matrix into the vapor phase [102].

Controlling such transformation is not trivial, therefore many approaches to water and wastewater monitoring have mainly used electronic tongues. This type of sensors may be based on two different effects, that is, *potentiometric* or *voltammetric* effects [127]. The main difference between the two types of array lies in the method which is used to determine the concentration of the analyte: specifically, with potentiometric e-tongues, the concentration of the analyte is assumed to be proportional to the potential between two electrodes, while, in voltammetric e-tongues, a voltage is applied between the electrodes, and the concentration of the analyte is computed as proportional to the measured current. Throughout the years, e-tongues have been mainly used in food industry, especially for wine classification [128]; however, they found application also to water quality monitoring [129].

Specifically, e-tongues have also been used for wastewater monitoring. As an example, in [126] a voltammetric array composed by eight metallic electrodes, capable of sensing gold, platinum, iridium rhodium, silver, copper, nickel and cobalt is used to determine and predict parameters measured in wastewater treatment plants. Another example is given in [137], were authors use both an e-tongue and high-performance liquid chromatography to forecast the value for pollutants found within wastewater deriving from detergents used in washing machines; results from both methods are compared, highlighting the slightly better performance achieved by the e-tongue. Furthermore, e-tongues have been exploited also in the development on innovative chemical techniques, such as flow-injection analysis [138], which allowed to quantify the concentration of nitrate ion in a water matrix without the treat the matrix itself by the removal of chloride.

Despite the aforementioned difficulties in the control of the transition between gas phase and liquid phase, and due to the poisoning effects to which e-tongues are subject, several efforts have been made to create electronic noses for water quality monitoring. One of the first approaches

is depicted in [139], where Gardner et al. describe a measurement system in which one of the main stages for the analysis is an electronic-nose made by six commercial odor sensors, each one based on the MOS effect described in chapter 2. Specifically, such system was used to monitor cyanobacteria over a period of 40 days and, due to several noise sources related to the setting, several preprocessing techniques were implemented in hardware to allow for a proper interpretation of data. However, results achievable by the system were satisfactory enough to allow the prediction of the different phases of the growth of cyanobacteria within water. Another approach was described in [140], where an electronic nose was also used to determine the presence within water of three different microbial species. In this case, a gas sensor array with 14 conducting polymeric sensors was used, and the results of the comparison between sterile water and water with traces of heavy metals such as arsenic, cadmium, lead and zinc were shown, highlighting that this type of system could be used to find either microorganisms or low concentration of heavy metals within different types of waters. In [141], a wastewater treatment plant was monitored with an electronic nose consisting of 12 metal oxide sensors. The monitoring campaign lasted 12 weeks, and both reference (that is, deionized water) and effluent were heated to 60 and 90 degrees to promote the volatilization and increase sensitivity. The main contribution of this work was in the development of both a *relative sensorial odour perception*, which expresses the correlation between the response of each sensor within the array, and the relative fingerprint of each substance. Finally, the possibility to deploy several e-noses throughout a water body, such as a lake or a basin, has been explored in [142], where an ESN made by several e-noses were deployed throughout the Riachuelo River, in Argentina, and the parameters acquired by each node of the network were compared to evaluate its status.

Still, several critical aspects of sensor arrays remains to be addressed. The first, important challenge is *selectivity*: as described by Nicolas et al. in [130], sensor arrays are limited in both *detection* (that is, the minimum value for the concentration of analyte which can be detected by the sensor) and *resolution* (defined as the minimum concentration needed to discriminate between two different analytes). Another issue is related to the drift of the sensors within the array: in [131], it is shown how humidity influences the response of QCM sensors, and underlines the need for a compensation model, which is achieved using a post-processing unit which embed

an ANN which has been previously trained on reference training data. Other approaches to drift compensation envisage for drift estimation using PCA [132]. Another important challenge lies in the standardization of methods, as actual implementations do not share common guideline in aspects such as data processing or number and typology of embedded sensors [32]. Finally, sensors are sensitive to several conditioning parameters, such as meteorological conditions, and may need recalibration even when the monitored environment changes [61].

To address some of the afore-mentioned issues, an integrated assessment platform, tailored for specific applications, has been proposed[63], and its usefulness has already been proved in several scenarios [64].

The next section will introduce a perspective on how data coming from sensor arrays are interpreted.

3.2 Data interpretation

From section 3.1, it is clear how gas sensor arrays generally require several precautions during their deployment. Once these challenges have been addressed, data can be processed. Current researches focus on two aspects: *preprocessing* and *algorithms* used for classification [31, 32].

Usually, preprocessing involves data conditioning, such as denoising and standardization. Also, feature extraction techniques may be used to obtain meaningful feature, possibly in complementary domains, such as frequency [33, 34].

Exploratory Data Analysis has also been used to properly understand data, for example by using polar plots[35]. However, if data are mapped into a high dimensional space (as an example, when the signature is composed by the readings of several sensors), visual exploration can be impossible; as a consequence, dimensionality reduction techniques are usually employed to perform an initial cluster analysis [37], remove redundancies [38], and allow for a simpler data visualization [39]. A widespread technique is PCA [36]; however, apart from it, feature selection, along with advanced data visualization techniques, such as t-SNE [40], may be suitable to be

used.

Afterwards, machine learning algorithms can be used to infer knowledge from data. Recalling the differences already depicted in chapter 2, data can either be considered as iid or time series.

In the first case, simpler approaches rely on the notion of distance between data points. That is, as each data point can be mapped to an n -dimensional space, with n equals to the number of feature, the distance between data points can be evaluated through a similarity index, as in [41]. Such approaches are similar to ranking procedures, where each data point can be ranked as closer or farther from the other one. Classification algorithms, need some kind of *boundary* between classes. Therefore, in [42] k-NN, which uses the Euclidean distance to first 'learn' about these boundaries, and the assign every new sample to one of these boundaries, is used for classification. k-NN is simple, and has good performance; however, as noted in [43], especially when n is high, the Euclidean distance may not be the best choice to compute distance between a couple of points. Other traditional statistical approaches are also used, such as *discriminant function analysis* [44, 45, 44] and *partial least squares* [46]. Artificial Neural Networks are also been employed for classification tasks in gas sensors [37]; more advanced approaches have used *ensemble learning* [47], with [96] which proposes the use of an *inhibitory SVM*[48]. Specifically, an ISVM trains one classifier f_i for each class i available within the dataset, and compares its output to the average output of the ensemble of classifier.

If data coming from gas sensor arrays are considered as time dependent, different algorithms are needed. A tool which has been widely used for such task are *Time-Delay Neural Networks* [49, 51, 50], and also recurrent neural networks have been employed for such tasks [52]. Another perspective has been given by [55], which proposes the use of *generative topographic mapping trough time* as an unsupervised model for time series inspection. Another interesting approach deals with the need to perform forecasting and prediction in real-time, which is especially useful in case of dangerous environments. Specifically, [53] provide predictions starting from data acquired in real time by using *reservoir computing* [54].

This section has given an overview on how data coming from sensor arrays are generally interpreted. In the next section, a particular focus will be given on wastewater data.

3.3 Water quality and wastewater

Wastewater are defined by Tilley [72] as follows:

> *(...) used water from any combination of domestic, industrial, commercial or agricultural activities, surface runoff or storm water, and any sewer inflow or sewer infiltration (...)*

Wastewater may be the outcome of a wide range of heterogeneous sources; hence, the characteristic of different effluents can be heterogeneous, and specific treatments must be used to restore water quality. Let us now introduce the concepts that should be used for an efficient evaluation of water quality indexes, therefore allowing for the implementation of a proper prevention and restoration intervention.

3.3.1 Wastewater identification and treatment methods

Possible sources of wastewater

The first step in the definition of a proper strategy of wastewater treatment lies into the identification of the source of the wastewater. Specifically, there could be two main possible sources:

- *urban wastewater* is composed by a mixture of black water (i.e. human excreta mixed with used toilet paper), gray water (i.e. washing water used by individuals for cars, dishes, or personal hygiene), and heterogeneous sources of domestic liquids (e.g. drinks, oils, paint, etc.) [74];
- *industrial wastewater* is composed by a wider range of compounds, including materials derived from industrial processes (e.g. site drainages, cooling or processing waters), organic wastes (either biodegradable, such as residuals from food production, or non-biodegradable, such as residuals from pharmaceutical manufacturing), toxic wastes, and many more[73].

There are also other possible sources of wastewater, such as agricultural wastewater, or residuals related to urban runoffs[74].

Treatments for wastewater

Once the source of the wastewater has been identified, a proper treatment strategy can be defined. Generally, the idea is to perform *water reclamation* [75], that is, treat the wastewater to make it again usable, with minimal risks.

One (desirable) precondition is to remove solid particles (e.g. mud, grit, etc.) as a pretreatment step, as they can easily compromise further processes. Afterwards, there are three levels of treatments:

- *primary treatments* aim to remove suspended solids, both organic and inorganic;
- *secondary treatments* aim to degrade biodegradable organics, therefore removing them through the intervention of biological processes such as bacterial digestion;
- *tertiary treatments* aim to chemically remove nutrients, toxic compounds, residual of suspended solids, and microorganism, using advanced techniques such as membrane filtration, percolation, active carbon and disinfection through chemical agents (such as chloride or ozone) or UV light.

Important characteristics for wastewater evaluation

The overall quality of the wastewater, both before and after the treatment, can be evaluated through standard methods and indexes[76].

First, one should evaluate *physical characteristics* of the wastewater. Specifically, temperature is important, as aquatic organisms (both fishes and plants) can survive only if the temperature of the water is within a certain range. Furthermore, high temperature may lead to sea warming [77], which a consequent replacement of indigenous species with alien ones. Another important

aspect is the presence of solids within water, which can be dissolved, suspended or settled as sediments. Turbidity is related to the fraction of suspended solids within the water, and can be evaluated by measuring the scattering of light which goes through the water; on the other hand, salinity, which influences the conductivity of the water, allows to determine the total dissolved solids.

Another important aspect are *chemical characteristics*. The first chemical characteristic is the concentration of ionized hydrogen, expressed by pH. Furthermore, one should evaluate the *dissolved oxygen*, which is important to sustain marine life, and *oxygen demand*, which can be either *biochemical* (BOD) or *chemical* (COD). Specifically, BOD measures how much oxygen is needed by bacteria and nutrients contained in the wastewater, while COD measures the demand related to reducing chemical within the matrix. Other chemical compounds which are usually monitored are *nitrogen*, which can be found in several forms and, being an important nutrient for plant growth, can contribute, in high concentrations, to eutrophication and algal bloom; *phosphate*, which are not toxic, but may be directly related to eutrophication [78]; and *chlorine*, a residual from bleaching and disinfection that can be harmful to animals [79].

Wastewater sampling procedure

The World Health Organization [80] depicts some guidelines to identify proper timing and location for sampling water destined to human consumption.

As for the location, samples should be taken from locations which are representative of important facilities or assets, such as a water source, treatment plants, storage facilities, and points where water is generally delivered or used. Furthermore, each of these location should be sampled individually. As for the minimum set of tests to perform, the most important (that is, microbiological quality, turbidity, free chlorine residual and pH) should be taken whenever a sample is taken. Finally, the guidelines suggest to perform a sampling operation whenever the 'situation demands', or when a 'change in environmental condition, outbreak of waterborne disease, or increase in incidence of waterborne diseases' occurs. However, these suggestions are quite generic, and, even if monthly sampling are suggested, data may not be sufficient to

perform a rigorous numerical analysis and, therefore, properly characterize the site. Furthermore, it is important to underline that these are only *suggestions*, and each country has its own commissioning authority to regulate water sampling; therefore, no global standard is currently available, and mathematical models cannot be therefore effectively generalized.

3.3.2 Wastewater Data Interpretation

Once data are acquired, they should be properly analyzed. It is important to underline that such analysis are often carried over time to assess the overall trend of water quality indexes.

In [81], Arya et al. use time series to perform univariate prediction on both dissolved oxygen and temperature for data acquired by four water quality assessment stations located near Stillaguamish River, in the state of Washington. Authors start from two consideration: first, univariate time series are long-memory processes, and, therefore, the present value is dependent on the values of the time series which lie several lags in the past; second, this type of series are rarely Gaussian distributed, and may not be successfully standardized. Therefore, the order series method is first used to standardize these series [82]; afterwards, authors deal with long-memory effects using FARIMA models [83]. Once the model is estimated, predictions are made, and evaluated using Pearson correlation coefficient, root mean square error and mean absolute percentage error. In [84], authors use ARIMA and Thomas - Fiering modeling [85] to forecast time series for water quality. Specifically, T-F model consider average monthly variations and correlation between data, and fit parameters into a set of n regression equations, where n is the number of time intervals (e.g. years, months, weeks, etc.) available within data. In the specific context of the dataset, authors conclude that all of the water flows analyzed showed seasonal patterns, probably due to the influence of annual cycles in the hydrological input to water streams, and no significant overall trends throughout the study period. In [86], authors perform an initial analysis through visual descriptors, which uses box-and-whisker plots for a visual description of possible trends within data. Then, seasonal Kendall test [87] is used to return a quantitative index. A multiplicative ARIMA model is then fitted [88], and both ACF and PACF are used to reveal seasonality. To evaluate the best fitted model, the

Akaike Information Criteria (AIC) [89] is used. Authors found that most of the stations do not show significant overall trends in water quality parameters, similarly to [84]; however, patterns show seasonal effects, and multiplicative ARIMA shows good fitting results. In [90], analysis are focused on *dissolved organic carbon* concentrations across the UK, along with flow, pH, alkalinity, air temperature and rainfall analysis. First, a seasonal Kendall test is used to give a quantitative index on these parameters. Then, a time series analysis, using ARIMA models, is performed. In this case, no clear evidence on the relationships between dissolved organic carbon and the aforementioned conditioning parameters are shown; however, trends show an overall increase in dissolved organic carbon, and several conclusions about relationships between increases in temperature and dissolved organic carbon can be made. This study can therefore be used for suggestions on concrete measures to take in both hydrological and climate change terms. In [91], four time series are taken from three catchments in the North and South of England, two near to agricultural catchments, one at the tidal limit, and one downstream of a sewage treatment works. ARMA models are used to evaluate nitrate levels, and predictions have been tested using standard RMSE, with an average percentage error below the 10% threshold. ARIMA and SARIMA modeling is also used in [92] to evaluate the concentration of boron in a specific test case; authors conclude that their approach is generalizable, thus ARIMA modeling is recommended for predicting the (univariate) boron concentration series within a generic river. In [93], one-month-ahead forecasts with transfer-function noise (TFN) [94] are combined with ANN, in a technique called *hybrid TFN+ANN*, to perform stream flow forecasting. Results indicates that this approach show improvements in generalization capability with respect to single TFN and ANN models.

Some of these techniques, along with the principles already described in chapter 2, will be used to describe numerical results in chapter 4.

Chapter 4

Experiments

In this chapter, the main contribution of this work is described.

The chapter will start with section 4.1, where an opportunistic sensing approach to cognitive radio in environmental sensor network is presented. Section 4.2 will then describe the development of the VPeN, an e-nose specifically tailored for water quality monitoring.

Afterwards, the approach to data analysis for three possible applications will be illustrated. It will start in section 4.3, where the datasets which have been used throughout this work are described. Then, in 4.4, results on the dataset acquired using the VPeN are shown, while in section 4.6 results of the analysis of time series acquired by two waste treatment plants are depicted. Finally, a multivariate approach based on the concept of complex network for the analysis of environmental data will be presented in section 4.7.

The last part, in section 4.8, will give a brief introduction to Env Lab, a tool which has been developed to perform numeric analysis on environmental data, and which has been made available as an open source project.

4.1 Opportunistic sensing approach to cognitive radio

Starting from the challenges highlighted in chapter 2, an approach to spectrum sensing for cognitive radio has been developed and presented in [11].

This approach has been lead by a simple design principle: that is, to guarantee good sensing performance, while keeping the related computational cost reasonably low. To this end, the method bases its foundations on the concept of *chirp signal*.

A chirp signal is a signal linearly modulated in frequency, and is described by the following:

$$c_n = e^{j\theta(n)} = c^{j\pi\alpha n^2} \tag{4.1}$$

From equation 4.1, it is possible to compute the instantaneous frequency of the signal, which is expressed by:

$$f_i = \frac{1}{2\pi}\frac{d\theta(t)}{dt} = \alpha t \tag{4.2}$$

The term α in equation 4.2 is the *chirp rate*. The instantaneous frequency spans the frequency axis on a band determined approximately by the following:

$$B_c = \alpha \cdot T_c \tag{4.3}$$

In equation 4.3, T_c is known as *chirp duration*. The value for α has to be defined according to the desired frequency resolution; it is important to underline that a high frequency resolution implies a lower chirp rate and, as a consequence, a longer observation time needed to span across all the bandwidth of interest.

In the time-frequency domain, the chirp signal is represented as a very narrow-band signal, linearly sweeping along a large bandwidth. A chirp can therefore be used to demodulate an

input signal to baseband, instead of the two-steps algorithm described in chapter 2. Specifically, the estimation is made multiplying the chirp signal for each input sample, and then applying a filter on the output signal, followed by an amplitude detector.

The scheme of the proposed algorithm is shown in figure 4.1.

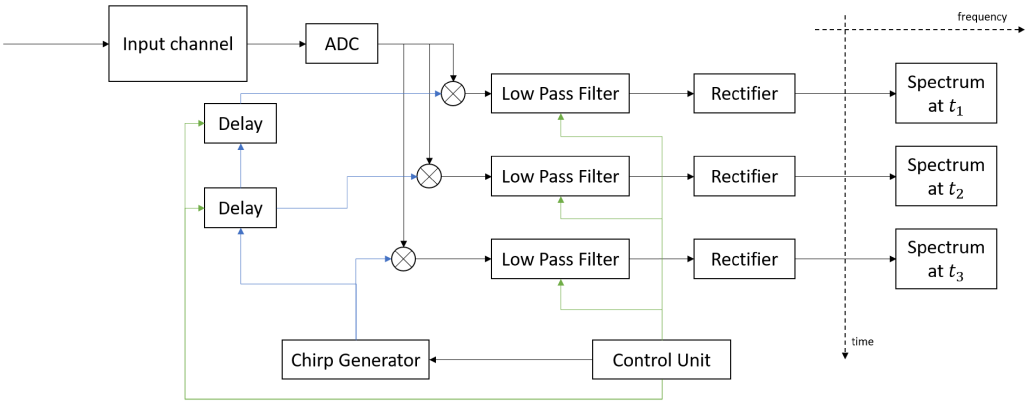


Figure 4.1: The block scheme of a possible implementation of the chirp-based spectrum sensing method.

The block scheme shows an example implementation of a chirp-based spectrum sensing method. Specifically, after the input signal is converted by an ADC, it is multiplied by a chirp at the required frequency, and then first low-pass filtered and, afterwards, goes through a rectifier, to give the spectrum at the given time instant. The estimation of the spectrum at delayed time instant can be achieved by simply using a delayed version of the chirp signal.

This scheme achieves two important goals: that is, each operation has a lower computational cost, if compared to the cost of the radix-2 FFT algorithm (see 2), and the demodulation of the whole signal can be easily parallelized.

To test the feasibility of the approach, simulations have been performed using an OFDM modulated signals with channelization of 1 MHz, and symbol duration of 1 ms, on a channel service bandwidth of 10 MHz, and the method has been compared with a reference implementation of a short-term Fourier transform. Results, which are extensively shown in [11], shows that, despite the lower computational complexity, the chirp-based demodulation scheme achieves results comparable to STFT in terms of SNR, and therefore is able to properly discern between the presence and the absence of the signal.

Thus, these encouraging results show that it is possible to implement this approach to allow for a better communication between distributed sensors in an ESN. This approach can be therefore taken into account for future deployment of a distributed architecture of VPeNs, whose working principles will briefly be described in the following.

4.2 Working principles of the VPeN

The architecture of the VPeN[102] is designed to be flexible, with each one of its components which can be interchanged with other, equivalent, devices, to upgrade, enhance and adapt its capabilities.

A working scheme of the VPeN is shown in figure 4.2.

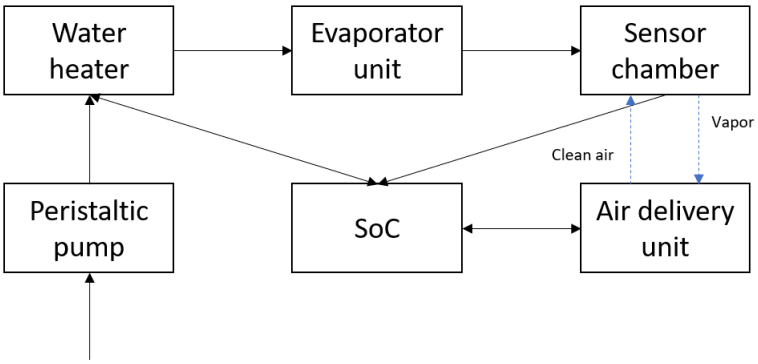


Figure 4.2: The block scheme of the VPeN.

At the core of the VPeN, there is a System on a Chip (SoC), which is responsible for the coordination of various parts of the instrument; furthermore, it collects, preprocess and store data coming from the measurement chamber, and handles all the tasks related to communication, storage and control.

A peristaltic pump injects the wastewater into the heater, which is responsible to regulate the temperature of the water matrix. An heating phase is needed to support the operation of the gas sensor array, as already described in chapter 2. To this end, the water heater uses a Peltier cell with three PID controllers, as described in [116]. The first PID controller operates in switch mode, interrupting the water flow to the first Peltier cell when it reaches a certain

threshold. The other two PID controllers are connected in series; the first heats the liquid, while the second cools it.

Once the water matrix has been heated, it is sent to the evaporator unit, which is able to vaporize the heated flow through impact. Vapor is then channeled to the sensor chamber, were measurement is performed; finally, an air delivery unity, based on an opposing fan system able to activate the measurement/cleaning cycles. Specifically, when the VPeN is in the measurement phase, air is pulled from the evaporator unit towards the external area, while in the cleaning phase external air is pushed towards the measurement chamber.

It is important to underline that the VPeN is tailored to allow the choice of sensors within the measurement chamber according to the specific application. The choice for the experiments which are the subject of this work will be described in the following section.

4.3 Datasets description

As already stated in chapter 2, the dataset is one of the main aspects which must be considered to perform a good data analysis.

To be effective, a methodology has to be tested against a significant amount of data, which must adequately model the real world: let us recall the example of ImageNet for image classification (cfr. chapter2).

However, acquiring a dataset such as ImageNet requires a lot of time and efforts. Therefore, it is desirable to *design* the acquisition of the dataset, using a multi-disciplinary approach, which involves both ICT and domain experts. Flaws in the design phase will lead to unpredictable results in data analysis.

And this will be the leitmotif of this section: how a good design helps in the interpretation of data, otherwise almost impossible to understand, disregarding the chosen machine learning algorithm.

4.3.1 The VPeN Dataset

The first dataset which has been acquired and evaluated will be referred to as *VPeN Dataset*. Data contained in this dataset are relative to two different use cases: the first one, called *IRSA Dataset*, has been acquired during the campaign described in [104] and [102], while the second one, called *ISMAR Dataset*, has been depicted in [98].

The IRSA Dataset

IRSA Dataset is described in [104], and has been acquired during the acquisition campaign of the MAUI experiment [102]. The IRSA Dataset contains a list of solutions selected to artificially resemble a set of compounds commonly found within urban and industrial wastewater. The campaign has been preceded by an extensive chemical study, which underlines the motivation behind the choice of the compounds [95]. Here, a brief, non-exhaustive overview of the solutions will be given.

Selected compounds and chemical considerations. Compounds have been analyzed in two different tranches, the first one made by the first four solution, while the second made by all the others.

The first three compounds which have been selected for the analysis are three salts dissolved in a water matrix. The first solution is composed by water and *sodium acetate*, the second by water and *ammonium bicarbonate*, while the third by water and *monobasic potassium phosphate*. Each one of these solutions has a single, fixed concentration, described in [95].

Chemical analysis performed on each one of these solutions states that[95]:

- the most relevant parameters found within solution 1 are *ammonia*, *nitrates*, *nitrogen*, *chloride*, *phosphorus*, *fluoride* and *phosphate*. Furthermore, the solution shows a relevant chemical-oxygen demand;

- the most relevant parameters found within solution 2 are *ammonia*, *nitrates*, *nitrogen*, *chloride*, *phosphorus*, *sulphates*, *fluoride* and *phosphate*;
- the most relevant parameters found within solution 3 are *nitrates*, *nitrogen*, *chloride*, *phosphorus*, *sulphates* and *phosphate*.

The fourth compound is made by a solution of all the above salts within a water matrix. The most relevant parameters found within solution 4 are *ammonia*, *nitrates*, *nitrogen*, *chloride*, *phosphorus*, *sulphates*, *fluoride* and *phosphate*, with a relevant chemical-oxygen demand.

Results are summarized in table 4.1.

Solution	Saline compound	NH_4	NO_3	N	Cl^-	P	SO_4	F	PO_4	COD
1	CH_3COONa	×	×	×	×	×		×	×	×
2	NH_4HCO_3	×	×	×	×	×	×	×	×	
3	KH_2PO_4		×	×	×	×	×		×	
4	All of the above	×	×	×	×	×	×		×	

Table 4.1: Results of chemical analysis performed on solutions 1-4 of IRSA Dataset

In the second tranche, solutions 5, 6, 7 and 8 have been analyzed. These solution were chosen to resemble compounds which can be commonly found within wastewater. In this case, however, each one of these solutions has been analyzed choosing three different concentrations within the water matrices. Again, the exact values for the concentrations are reported in [95].

Solution 5 is composed by *soya peptone* dissolved in a water matrix. Chemical analysis reveal the following:

- an increment in *ammonia*, *nitrogen*, *phosphorus*, *sulphates*, *fluoride* and *phosphates* as the concentration of soya peptone increases;
- a relevant chemical-oxygen demand.

Solution 6 is composed by *starch* dissolved in a water matrix. Chemical analysis reveal the following:

- an increment in *ammonia*, *nitrogen*, *phosphorus*, *sulphates* and *fluoride* as the concentration of starch increases;
- a relevant chemical-oxygen demand.

Solution 7 is composed by *milk powder* dissolved in a water matrix. Chemical analysis reveal the following:

- an increment in *nitrogen*, *chloride*, *phosphorus*, *sulphates*, *fluoride* and *phosphates* as the concentration of milk powder increases;
- a relevant chemical-oxygen demand.

Solution 8 is composed by *yeast extract* mixed with a water matrix, and chemical analysis reveals the following:

- an increment in *ammonia*, *nitrogen*, *chloride*, *phosphorus*, *fluoride* and *phosphates* as the concentration of yeast extract increases;
- a relevant chemical oxygen demand.

The last solution under analysis is solution A, given by a mixture of all previous solutions in a water matrix. Three different settings for the concentrations of the solutions were used, mainly by increasing the concentration of compounds belonging to the second tranche, while keeping the concentration of compounds belonging to the first tranche stable. Chemical analysis reveal the following:

- a *decrement* in *ammonia* as the concentration of compounds 5, 6, 7 and 8 increases;
- an increment in *nitrates*, *nitrogen*, *chloride*, *phosphorus*, *sulphates*, *fluoride* and *phosphate* as the concentration of compounds 5, 6, 7 and 8 increases;
- a relevant chemical-oxygen demand.

Results are summarized in table 4.2.

Solution	Saline compound	NH_4	NO_3	N	Cl^-	P	SO_4	F	PO_4	COD
5	Soya peptone	×	×			×		×	×	×
6	Starch	×	×			×	×	×		×
7	Milk powder		×		×	×	×	×	×	×
8	Yeast extract	×	×		×	×			×	×
A	All compounds	×*	×	×	×	×	×	×	×	×

Table 4.2: Results of chemical analysis performed on solutions 5-6-7-8-A of IRSA Dataset. *It is important to underline that the chemical analysis shows that ammonia *decrements* when the concentration of compounds in solution A increases.

ISMAR dataset description

ISMAR Dataset is described in [98], and has been acquired during another sampling campaign, conducted in the aquaculture plant located in Ravenna, Italy. Specifically, three samples were taken:

- solution B is made up by sea water taken from a water tank for aquaculture before the insertion of mussels;
- solution C resembles artificial sea water;
- solution D is made up by sea water taken from a water tank after five hours from the insertion of mussels.

As for the water tank, its volume is of 1500 liters; the quantitative of mussels which are inserted into the tank is of about 200 kilograms. Finally, water inside the tank gets replaced by external water through a pump.

In this case, tests are not carried with different concentration, but, instead, with different heating temperatures (specifically, 30, 45 and 60 Celsius).

No further chemical analysis have been carried on samples. However, it is important to underline that, as depicted in [99], it is expected that this experiment highlights the contribution of bivalves to methane and nitrous oxide fluxes within the water matrix.

Acquisition settings

The measurement chamber of the VPeN has been set up as described in table 4.3.

Port	Sensor	LPG	H_2	CH_4	C_3H_8	$-OH*$	NH_4	CO	CO_2	G**	E***	C_7H_8
1	MQ 2	×	×		×							
2	MQ 3					×				×		
3	MQ 4	×		×								
4	MQ 5	×	×	×								
5	MG 811								×			
6	MQ 8		×			×						
7	MQ 7		×					×				
8	MQ 9	×		×				×				
9	MQ 6	×		×								
10	MQ 137		×				×				×	
11	MQ 135		×				×					×

Table 4.3: Gas sensors in the measurement chamber of the VPeN, along with sensed substance. *Alcohol **Gasoline ***Ethanol

This configuration has been used for the acquisition of both the IRSA and the ISMAR datasets. Obviously, the main focus of this configuration is in the capability to discern organic compounds, along with hydrogen, carbon dioxide and ammonia.

By comparing the compounds which have been found in the analysis carried in [95], one should expect that the sensors which should be more discriminative should be the ones which can sense ammonia for IRSA Dataset, and nitrogen/methane for ISMAR Dataset.

The period used for sampling is of two seconds, and each measurement cycle lasts 300 seconds; therefore, there are 150 samples per measurement cycle. Afterwards, a cleaning cycle is performed, whose duration can vary according to specific needs, as designed by the hardware producer and maintainer.

It is important to underline that this procedure does not envisage for an initial 'zero calibration' stage, diverging from the methodology described by [53]. It is therefore important to normalize data in a post-processing step to compare results from different experiments.

It is also important to consider the experimental conditions: specifically, it has not been used a laboratory with a strictly monitored equipment, but results were directly recorded on the

field. Therefore, experiments were performed in suboptimal conditions, with inadequate ventilation and thermal control; this has lead to noisy data. Furthermore, measurement cycles were characterized by non-constant transients, which have not been characterized by the produced. Therefore, due to this chaotic behavior, an experimental threshold has been used, removing the first 30 seconds of each measurement cycle. This value has been set experimentally using the average time in which an 'elbow' was found within the temporal plots of the sensors, and may be susceptible of variations in the future.

4.3.2 Gas Sensor Array in Open Settings

This dataset, which has been acquired by Vergara et al. and described in [96], has been used for mainly two reasons. First, it is less noisy than the VPeN Dataset; second, it offers enough data to model 72 different sensors, therefore allowing to test a multivariate methodology based on complex networks. The dataset is freely available at the UCI Machine Learning Repository [97].

The Gas Sensor Array in Open Settings dataset holds data acquired by a set of nine identical electrical noses, each composed by eight different MO-X sensors. The selected sensors belong to the TGS26-XX family, and are sensitive to hydrocarbons, hydrogen, nitrogen, sulfur compounds, and carbon monoxide.

Data are acquired in a wind tunnel test-bed facility, where the electrical noses were positioned at six different locations, namely $L_1, \ldots, L_6$, normal to the wind direction, and uniformly distributed throughout the tunnel. At each trial, a different chemical compound is injected within the tunnel, with a specific concentration. Specifically, these chemical compounds are acetaldehyde, acetone, ammonia, benzene, butanol, carbon monoxide with two different concentrations, ethylene, methane, methanol and toluene. Let us note that, in contrast with what happens in the VPeN dataset, each one of these substances are characterized by high volatility. Furthermore, the choice of the sensors appears to be well-suited to respond to the selected compounds.

The entire reference-measurement-cleaning cycle lasts about 260 seconds. Throughout the

trials, two different conditioning parameters are considered. The first is the *heater voltage* V_h, which is directly related to the temperature at the active surface of each sensor, while the second is the *airflow speed* S, i.e. the speed of the fan within the test-bed. For V_h, five values are considered, while only three were considered for S. Specifically, allowed values for V_h are $\{4.0, 4.5, 5.0, 5.5, 6.0\}$, while allowed values for S are $\{0.10, 0.21, 0.32\}$.

4.3.3 IRSA Wastewater

This dataset contains samples gathered from two different wastewater treatment plants, located in Monza and Vimercate, next to Milan, Italy.

Sampling has been performed across a time span of about two years, spanning from January 2016 to October 2017.

Several types of parameters were monitored; however, just a subset of them was considered for analysis, as explained in [95]. It is also important to note that sampling was not performed on a regular basis, therefore some data preprocessing has been needed to apply time series modeling algorithms.

4.4 Results on the VPeN Dataset

The goal of this analysis is to evaluate the goodness of data acquired by the VPeN, and, hence, its suitability to be used as a device for real-time wastewater quality monitoring. A protocol to evaluate the correspondence between chemical analysis and numerical results has therefore been established, along with some guidelines which should lead future deployment of networks of such sensor.

The protocol is directly established from the following considerations:

- *data can be labeled according to the chemical analysis*, that is, the knowledge of which substance is being analyzed automatically allows the experimenter to set some labels

suitable for supervised learning;

- *experiments are executed on field*, and therefore in suboptimal conditions. Hence, a way to evaluate how achieved results fit the expected ones is needed.

To this end, some of the ideas presented in [109] have been borrowed. Specifically, a clustering algorithm has been used to verify the fitness between the data distribution in the feature space, and the given assignment of labels. This also gives several hints on how to improve the design of the experimental phase, processing data to make them suitable for real-time monitoring.

4.4.1 Experimental settings

Selection of the clustering algorithm

The clustering algorithm has been selected according to two different factors.

The first one is related to the distribution of data. In fact, many clustering algorithms, such as K-means [144], require data to follow a specific distribution, which is often a normal distribution, or even a specific shape in the feature space. As for the first hypothesis, a one sample Kolmogorov-Smirnov test on VPeN dataset confirmed that data within it do not follow a normal distribution.

The second factor is related to one of the most important hyper parameters needed by several clustering algorithms, that is, the number of clusters which are expected to be found within data. As already stated, these experiments are meant to be *data driven*, therefore the algorithm should automatically derive the number of clusters from data themselves.

Given these factors, DBSCAN [143] has been chosen as clustering algorithm. DBSCAN bases its clustering on *density*: clusters are defined as areas of highly dense data points, separated by areas with low density. As a consequence, it is not required that data have a predefined distribution or shape.

An important concept on which DBSCAN is built upon is the concept of *core samples*. A core sample is a sample such that exists a number of *min-pts* other data points within a distance ε from the sample itself; these points are defined as *neighbors* of the core sample, and may be core samples as well. DBSCAN operates recursively, by taking a first core sample, evaluating all its core samples within a range ε, and then evaluating its core samples as well, and so on, therefore building clusters from zones with high density. DBSCAN also includes a mechanism to found outliers, which are defined as non-core samples which are at a distance above ε from any other core sample.

Evaluation of clustering performance

Another important thing that has been considered in this work is the metric used to evaluate clustering performance. The main idea behind this choice is that the clusters found by DBSCAN should both *resemble the labeling* and be *well separated into the feature space* (that is, not overlapped). Therefore, two metrics have been selected, that is, *Adjusted Rand Index* [110] and *Silhouette Score* [111].

Adjusted Rand Index is a function of the similarity between a given labeling and the results achieved by the clustering algorithm, and is expressed as follows:

$$ARI = \frac{RI - E[RI]}{max(RI) - E[RI]} \tag{4.4}$$

In the previous equation, RI is the *Rand index*, expressed as:

$$RI = \frac{a + b}{C_2^{n_{samples}}} \tag{4.5}$$

and:

- C is the ground truth;

- a is the number of pairs of elements that are in the same set in C and in the same set in the given clustering K;
- b is the number of pairs of elements that are in different sets in C and in different sets in the given clustering K.

Silhouette Score allows instead to define how 'separated' clusters are in the feature space. Specifically, it is defined as:

$$s = \frac{d - c}{max(d, c)} \tag{4.6}$$

Where:

- d is the mean distance between a sample and all other points in the same class;
- c is the mean distance between a sample and all other points in the nearest cluster.

In the following, the application of these two scores will be properly described.

Hyper parameter selection

It must be underlined that DBSCAN still requires two hyper-parameters, that is, the values for *min-pts* and ε. However, due to the large quantity of data which should be evaluated, a method to automatize the selection of both these hyper-parameters through grid search [145] has been developed.

However, the range for the values to be searched have been set according to some common rules of thumbs used to set hyper-parameters in DBSCAN.

First, the value chosen for *min-pts* is directly related to the number of dimensions of the dataset [105, 106], and lies within the range $[d + 1, 2 \cdot d]$, where d is the number of dimensions of the dataset. Furthermore, in case of noisy data, the authors suggest to select a value larger than

$2 \cdot d$. As for the value of ε, it can be determined using a k-distance graph [105, 106], and choosing an optimal elbow from the plot of the distances between the k-nearest points. Obviously, with noisy data, choosing a higher value for *min-pts* is desirable, due to the fact that data are more scattered and, as a consequence, high density areas are likely to be more spread throughout the feature space.

These rules of thumbs have lead to select the following values for grid search:

- the lower admissible value for *min-pts* has been set to $(min - pts)_l = d + 1 = 12$, since the VPeN dataset is supposed to have 11 dimensions (a dimension per sensor, excluding port 8 which does not hold any sensor in the experiments); the higher admissible value has set to $(min - pts)_h = 3 \cdot d = 36$, to take into account noisy data;

- a k-nearest neighbor graph obtained using data from all the experiments has been evaluated for both $k = 12$ and $k = 36$, obtaining an approximate values for the elbows of $\varepsilon_l = 0.02$ (for the lower admissible value) and $\varepsilon_h = 0.8$ (for the higher admissible value).

Description of the experiments

The experiments can be described as follows. First, the impact of the *conditioning parameters* (that is, either the heating temperatures or the concentration of the solution) has been evaluated. The idea is to determine whether the instrument is able to discriminate when such parameters vary. However, if it is intuitive that a correct evaluation of the concentration of the substance is relevant to quality monitoring, assessing the discriminative power of the device to different temperature may not be so obvious. Let us recall the device which has been proposed by Dewettinck [141], whose working principle is very similar to the one on which VPeN is based. Specifically, the e-nose proposed by Dewettinck uses different temperatures to facilitate the vaporization of the solution, specifically 30, 60 and 90 Celsius degrees. Therefore, the question that one may ask is: *how does different temperature influence the response of the sensors*? Answering this question may be useful to determine the narrow the range of temperature in which

the VPeN may operate, therefore improving its sensitivity, and either narrowing or widening the range of sensed compounds.

Another important aspect that has been evaluated is the capability of the VPeN to discriminate between two (or more) different solutions. To this end, the signature acquired by the VPeN for the solutions within each tranche have been compared. Obviously, if n solutions are compared, one should expect exactly n (hopefully well-separated) clusters.

Furthermore, as experiment had the possibility to evaluate the response of two VPeNs, the coherence and, therefore, the repeatability of the measurements have been tested. Intuitively, since the instruments are supposed to be identical (that is, with the same set of sensors in the measurement chamber, and exposed to the same environmental setting), one should expect that results for the same solution on different VPeN should belong to the same data generation process and, after proper normalization, roughly to the same cluster.

For each solution, two pair of values for both ε and *min-pts* have been reported in the results. The first pair is the one which guarantees the best possible value for the ARI; the second is relative to the best possible value for the silhouette score. One may argue that just one value for ε and *min-pts* can be chosen, combining both scores in a single metric; however, in this work, the two aspects have been kept distinct, even if it is reasonable that, in optimal conditions, the values for ε and *min-pts* should be almost equal, as cluster should be both well separated and maximally resemble the ground truth.

Finally, the percentage of outliers found by each clustering procedure has been reported. As already depicted when describing the algorithm on which DBSCAN is based, outliers are directly related with noise, and can help to identify whether there is an alert (if the number of outliers is relatively low, that is, anomalous situations are not common, and are an effective indicator that something unusual is happening in the water matrix) or if the measurement chamber is unsuited for the specific wastewater composition (i.e., if the number of outliers is relatively high, it is possible that either sensors are poisoned, or that they are sending chaotic, non coherent responses due to saturation, drift, or values below the instrumental threshold).

Results are presented in tabular form. For each solution, performance scores and number of clusters will be reported, along with the theoretical number of clusters expected from the knowledge of the ground truth. To guarantee for the repeatability of the results, values selected by the grid search algorithm for both ε and *min-pts* are also given.

Feature importance with random forests

Random forests have been introduced by Breinman in [146]. These algorithms are based on an ensemble of decision trees [147], where each node in the tree is a condition on a single feature which allows to *split* the dataset, therefore causing similar responses to end up within the same split. Random forests also provide two methods for feature selection, that is, *mean decrease impurity* and *mean decrease accuracy.*

The *impurity* in a random forest is a measure on which the local optimal condition is chosen; typical choices for measuring impurity are either gini index or information gain criteria [148]. Intuitively, when a tree is being trained, it is possible to compute how much each feature decreases the weighted impurity in a tree; for the whole forest, the impurity decrease from each feature can be averaged, and the features can be ranked accordingly. It is extremely important to note that impurity is biased towards variables with more features [149], and that if the dataset has two or more correlated features, these are interchangeable from the point of view of the model. Another method is to evaluate the decrease in the accuracy, iteratively permuting the set of features used for classification, and evaluating the impact of this permutation on the accuracy of the model.

In this work, the mean decrease impurity has been used, as variables within the dataset have the same number of features. To address the effects which may be related to the correlation between variables, a correlation analysis is performed on relevant features, to ensure that these are not correlated and, therefore, effectively relevant. Finally, features are considered to be relevant if their relative score, in terms of mean decrease impurity, is above the relevance threshold of 25% (that is, they account for at least a quarter of the variations which can be found within data).

It is important to note that, in many cases, data have been found too noisy by the implementation of the random forest classifier, even with the optimal settings suggested in [150], that is, fully developed trees, a high number of trees in the forest (higher than the number of sample itself), and a number of maximum considered features set to d, where d is the number of features within the dataset (i.e. the sensors in the measurement chamber). Therefore, only the combinations which have been successfully been analyzed by the random forest have been reported.

4.4.2 Results on IRSA Dataset

The first results which are reported have been achieved on the IRSA Dataset. Let us briefly note that, as for the ARI, it is not reported for solution 4, as data are supposed to belong to one expected cluster. Therefore, recalling the definition given in section 4.4.1 for the ARI, it is clear that both term a and b need more than one cluster in the labels to be defined, and, in this case, this condition is not met.

In the following, results will be first described for each VPeN (in sections 4.4.2 and 4.4.2), and then for data coming from both the instruments to evaluate for repeatablity (in section 4.4.2).

This protocol will be also used to report results for the ISMAR Dataset in section 4.4.3.

VPeN 11

Single solutions. Let us start by analyzing table 4.4, which describes the best ARI for the single solutions under analysis.

Let us start by analyzing the results on the first tranche. For solutions 1 and 2, the best ARI is low, and the number of clusters which have been found considerably differs from the expected values. Also, the percentage of outliers is relevant; this suggests both environmental noise (due to the high number of outliers) and suboptimal acquisition settings (due to the low ARI). Solution 3 shows a higher ARI and a lower number of outliers, suggesting a reduced impact of

Solution	ARI	ε	*min-pts*	Clusters	Exp. clusters	Outliers (%)
1	0.29	0.8	12	10	3	20.89 %
2	0.22	0.8	12	11	3	18.2 %
3	0.70	0.8	12	6	3	3.78 %
5	0.84	0.78	12	4	3	2.89 %
6	0.92	0.8	12	4	3	2.2 %
7	0.96	0.8	12	3	3	2.44 %
8	0.64	0.76	12	8	3	3.11 %
A	0.33	0.8	12	7	3	7.77 %

Table 4.4: Best Adjusted Rand Index for VPeN 11 on single solutions of IRSA Dataset

environmental noise and a better suitability of the selected acquisition settings.

As for the second tranche, solution 5, 6, and 7 show a high value for the ARI, while solution 8 and especially solution A show a considerably lower value. However, as the percentage of outliers is low, the suggestion is that either the acquisition settings are biased, or there are conditioning effects due to environmental settings which repeat themselves and, as a consequence, can be modeled and removed. However, such evaluation require a proper knowledge on the environmental setting, which is not available within the VPeN Dataset.

In table 4.5 the best silhouette score for the experiments on single solutions is shown.

Solution	Silhouette	ε	*min-pts*	Clusters	Exp. clusters	Outliers(%)
1	0.18	0.8	13	9	3	24.3 %
2	0.08	0.68	13	13	3	29.78 %
3	0.49	0.78	17	8	3	6.2 %
4	0.25	0.8	14	4	1	26.3 %
5	0.48	0.66	19	6	3	10.44 %
6	0.61	0.27	30	2	3	69 %
7	0.56	0.8	17	4	3	5.44 %
8	0.59	0.76	17	7	3	8.33 %
A	0.15	0.59	12	12	3	16.33 %

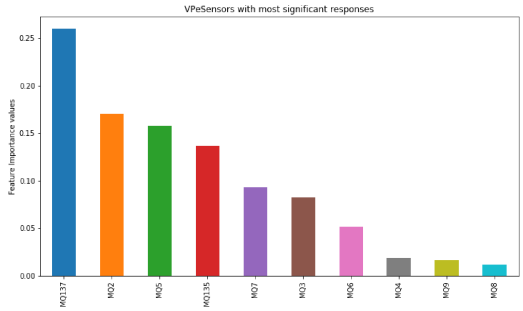
Table 4.5: Best Silhouette Score for VPeN 11 on single solutions of IRSA Dataset

Interestingly, the values of ε and *min-pts* for which the best possible silhouette score is achieved are similar to the ones obtained for the best possible ARI. However, silhouette score is, on average, low; this suggest that, even if the instrument is capable to return results which adhere to the assigned ground truth, most of the clusters found within data are overlapped. This, again, suggest uncontrolled biases in the acquisition and environmental settings, which result

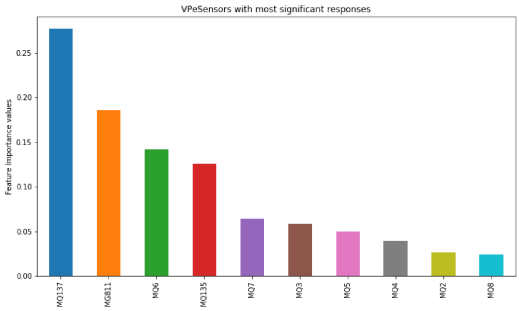
in chaotic data and overlapped clusters.

Let us now show results when a feature selection is performed through random forest.

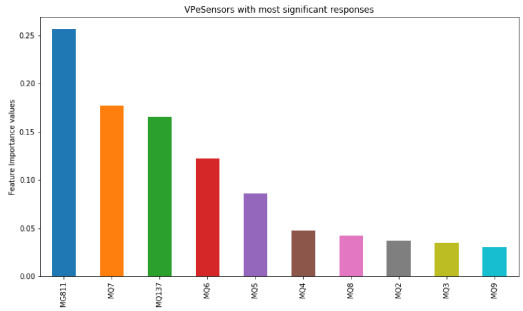
Important features for single solutions. In figure 4.3, the results of the features ranked by a random forest classifiers on solutions 2, 6, 8 and A are shown. As already stated, these are the only solutions on which a random forest classifier, with the optimal settings suggested in [150], could work.



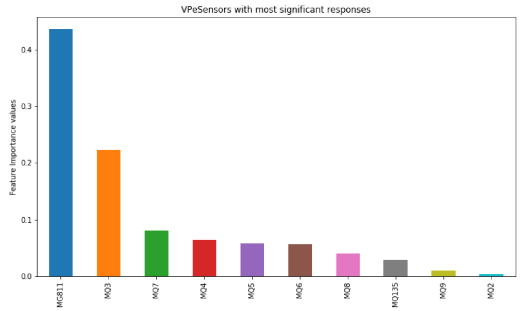
(a) Solution 2



(b) Solution 6



(c) Solution 8



(d) Solution a

Figure 4.3: Features ranked according to their relevance for VPeN 11 single solutions

The most discriminative sensors, according to the conditioning parameters (which are temperature for solution 2, and concentration for solutions 6, 8 and A) are:

- for solutions 2 and 6, sensor MQ 137;
- for solutions 8 and A, sensor MG 811.

Let us now briefly recall table4.3, which states that:

- the most discriminative sensor for solutions 2 and 6 (MQ 137) can sense ammonia, hydrogen and ethanol;
- the most discriminative sensor for solutions 8 and A (MG 811) can sense carbon dioxide.

Recalling the experiments performed in [95], it can be found that:

- solution 2 decomposes into the water matrix into ammonia and carbon dioxide, and one of the expected compounds is (indeed) ammonia;
- solution 6 decomposes into the water matrix into several compounds, one of which is ammonia;
- solution 8 is composed by yeast extract which, in absence of oxygen, are subject to a process called *fermentation* [95], which produces carbon dioxide and ethanol;
- solution A simulates an urban wastewater.

Given that, the following can be deduced:

- the highly-discriminative response of the MQ 137 sensor to solution 2 is related to a different quantity of particles of ammonia which are released when the temperature of the water heater changes. Therefore, one of the working principle of the VPeN, that is, pre-heating the solution facilitates the release of compounds of interest, is confirmed, and different behaviors correspond to different temperatures;
- the highly-discriminative response of the MQ 137 sensor to solution 6 extends previous consideration to the situation where the concentration of the solution varies. Hence, the VPeN can be used, if properly set, to discriminate between increasing concentrations of compounds which emits ammonia when dissolved into the water matrix, therefore allowing one to use it for alert detection;

- the highly-discriminative response of the MG 811 sensor to solution 8 is related to a different concentration of yeasts within the water matrix, and, therefore, to a different impact of the fermentation phenomena. Previous consideration on solution 6 can therefore be extended to solution 8;

- the highly-discriminative response of the MG 811 sensor to solution A is probably related to complex interactions between organic matter and bacteria found within wastewater. These are likely to produce carbon dioxide [95], therefore considerations shown for solution 6 and 8 are confirmed.

Let us evaluate how performance change when only a reduced set of features is used for clustering.

In table 4.6 the best ARI achieved for the aforementioned solutions when only relevant features are selected is shown. It is interesting to note how, if compared with the analysis with the full feature set, there is an improvement for solutions 2 and A, but results on solutions 6 and 8 are deteriorated. This suggest that, in this case, the value set for the relevance threshold (that is, 0.25) is too high. Hence, for these two solutions, analysis have been carried out lowering the threshold value to 0.15, and are also reported in tables 4.6 and 4.7.

Solution	ARI	ε	*min-pts*	Clusters	Expected clusters	Outliers (%)
2	0.37	0.18	13	5	3	3.4 %
6	0.56	0.25	12	3	3	0.22 %
6*	1	0.72	12	3	3	0 %
8	0.49	0.02	21	11	3	7.55 %
8*	0.87	0.55	12	4	3	0 %
A	0.63	0.16	12	3	3	0 %

Table 4.6: Best Adjusted Rand Index for VPeN 11 on single solutions of IRSA Dataset when most important features are selected. *Results with relevance threshold lowered to 0.15.

As it can be seen, lowering the relevance threshold to the experimentally-found value of 0.15 helps to achieve improved results, with a perfect match for solution 6, and a considerable enhancement for solution 8. However, results in terms of silhouette score, which are reported in table 4.7, show a deterioration when this threshold is lowered. This may be acceptable for the specific application purposes, as a perfect match with the ground truth is achieved, but

should be further investigated, by improving both environmental and acquisition settings, to give a proper explanation of this phenomena; as often happens in these cases, more data are needed to give further suggestions..

Solution	Silhouette	ε	*min-pts*	Clusters	Expected clusters	Outliers (%)
2	0.91	0.06	12	10	3	4.0 %
6	0.96	0.24	24	3	3	4.0 %
6*	0.42	0.71	12	3	3	0 %
8	0.86	0.22	12	2	3	0 %
8*	0.74	0.25	26	6	3	0 %
A	0.89	0.12	14	4	3	0 %

Table 4.7: Best Silhouette Score for VPeN 11 on single solutions of IRSA Dataset when most important features are selected. *Results with relevance threshold lowered to 0.15.

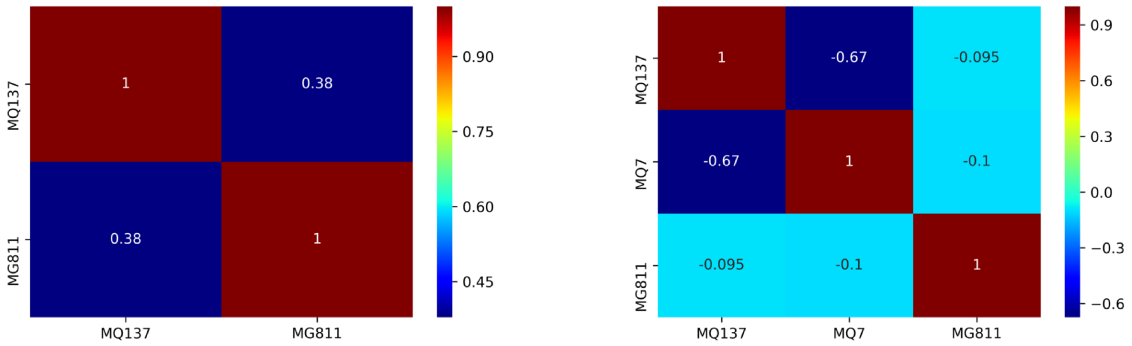
It is important to briefly examine the effects of a lowered relevance threshold on solutions 6 and 8.

For solution 6, sensors which are found to be relevant with the lowered threshold are MQ 137 and MG 811. This suggests that, as the concentration of starch increases, the release of carbon dioxide slightly varies, therefore allowing a proper discrimination between different solutions.

For solution 8, sensors which are found to be relevant with the lowered threshold are MQ 7, MQ 137 and MG 811. This reinforces the consideration that a fermentation occurred within the water matrix, and that the release of the products of such a chemical reaction are the ones which have been read by the VPeN: in fact, sensor MQ 137 can sense another product of fermentation, that is, ethanol [95]. The relevance of sensor MQ 7 should be examined in depth with the help of a domain expert.

For both solutions, a correlation analysis of the responses of the most relevant sensors is shown in 4.4. Correlations have been computed using the Kendall's τ, as monotonicity and linear relationships between responses cannot be supposed.

From figure 4.4, it can be underlined that the responses of these sensors are not correlated over time (but, instead, the responses of MQ 7 and MQ 137 on solution 8 are anti-correlated). Hence, it is possible to conclude that each one of these sensors has a role in the discrimination between different concentrations of solutions 6 and 8, and therefore should be considered in an



(a) Solution 6 (b) Solution 8

Figure 4.4: Correlation analysis for the responses of most relevant sensors for solutions 6 and 8.

optimal acquisition setting.

Let us now evaluate how readings from the VPeN are fit to distinguish between several substances. It is important to underline that *this is not a classification*, therefore the final goal is not to establish whether the VPeN is able to classify different solutions, but, instead, to have some hints about undergoing processes which bias the data acquisition (and, as a consequence, to have a perspective on how to remove such effects).

Multiple solutions. In tables 4.8 and 4.9, results of the comparison between multiple solutions for the VPeN 11 are shown. In this case, the effects of the variation of the conditioning parameter, along with the variation of the solution, must be considered.

Let us start with table 4.8. Results show that the adjusted rand index is low for most of the comparisons, except for solutions 1-3 and solutions 5-6-7-8.

Solutions	ARI	ε	*min-pts*	Clusters	Exp. clusters	Outliers
1-2	0.36	0.72	19	13	6	19.4 %
1-3	0.62	0.76	12	13	6	7.06 %
2-3	0.42	0.8	18	11	6	7.33 %
1-2-3	0.40	0.8	13	12	9	7.78 %
1-2-3-4	0.41	0.51	32	13	10	19.16 %
5-6-7-8	0.76	0.70	12	13	12	2.22 %

Table 4.8: Best Adjusted Rand Index for VPeN 11 on the comparison of multiple solutions on IRSA dataset

As for the best silhouette score, it is shown in table 4.9. Results are also poor, and suggest that clusters are overlapped in the feature space.

Solutions	Silhouette	ε	*min-pts*	Clusters	Exp. clusters	Outliers
1-2	0.20	0.70	12	13	6	14.72 %
1-3	0.51	0.8	17	10	6	9.17 %
2-3	0.45	0.78	13	11	6	6.83 %
1-2-3	0.47	0.8	13	12	6	7.78 %
1-2-3-4	0.45	0.8	12	8	10	1.53 %
5-6-7-8	0.26	0.8	30	13	12	3.5 %

Table 4.9: Best Silhouette Score for VPeN 11 on the comparison of multiple solutions on IRSA dataset

It is important to underline how, on average, the number of outliers is low; however, especially the low ARI suggest that, for multiple solutions, there are several bias effects which should be considered.

Suggestions. Results on VPeN 11 highlight the following: on one hand, in several cases, the acquisition settings, that is, the choice of the sensors which have been embedded in the measurement chamber, is not adequate to the specific use case scenario. On the other hand, except for some relevant cases, data do not appear to be excessively noisy and, therefore, experimental settings may be accepted. Therefore, the suggestion may be to refine the selection of sensors to better fit them to the use case. Let us now proceed with the evaluation of the results coming from the other instrument, named VPeN 12, which is supposed to be identical to VPeN 11. The experimental protocol which has been followed is the same followed for the VPeN 11.

VPeN 12

Single solutions. In table 4.10, the best ARI achieved for single solutions on VPeN 12 is shown. For the first tranche of solutions, results are on average slightly better than the ones achieved by VPeN 11. However, for the second tranche, results are considerably worse.

In table 4.11, the best silhouette score is shown for the same solutions. These results are more

Solution	ARI	ε	*min-pts*	Clusters	Exp. clusters	Outliers
1	0.58	0.76	12	4	3	2.89 %
2	0.37	0.8	12	11	3	16.2 %
3	0.72	0.76	12	5	3	3 %
5	0.48	0.8	12	7	3	7.12 %
6	0.76	0.8	12	5	3	5.33 %
7	0.53	0.8	12	6	3	4 %
8	0.37	0.8	12	10	3	15.56 %
A	0.37	0.8	12	10	3	12.11 %

Table 4.10: Best Adjusted Rand Index for VPeN 12 on single solutions of IRSA Dataset

in line with the ones achieved by VPeN 11, even if there is an anomalous situation for solution 2, where the best possible silhouette score envisage for an extremely high number of outliers.

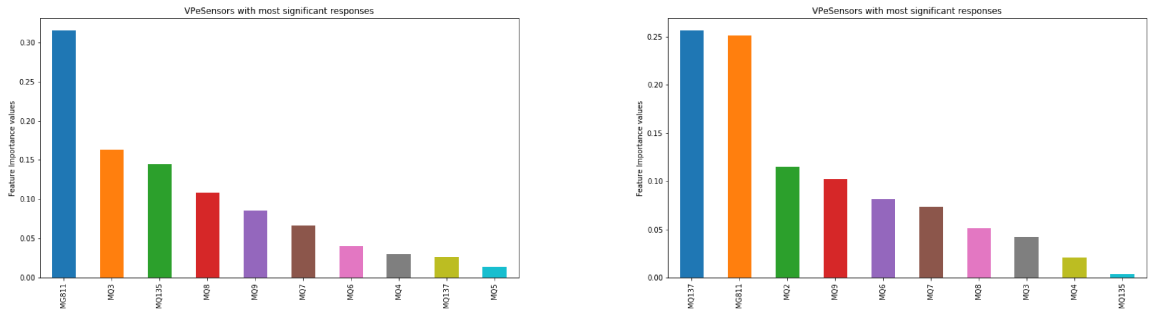
Solution	Silhouette	ε	*min-pts*	Clusters	Exp. clusters	Outliers
1	0.62	0.8	26.6	3	3	6.67 %
2	0.29	0.02	17	1	3	97.56 %
3	0.63	0.66	12	6	3	3.77 %
4	0.34	0.72	13	4	1	36.67 %
5	0.45	0.8	12	7	3	7.12 %
6	0.66	0.8	12	5	3	5.33 %
7	0.54	0.8	13	7	3	4.33 %
8	0.33	0.8	12	10	3	15.56 %
A	0.43	0.8	13	10	3	12.55 %

Table 4.11: Best Silhouette Score for VPeN 12 on single solutions of IRSA Dataset

Overall, the instruments achieve similar results, which is expected due to the fact that they are exposed to the same exact environmental conditions, and have the same acquisition setting. Furthermore, slight differences are expected to be found, due to complex drift phenomena, and different usage and/or poisoning of different sensors. However, data acquired by VPeN 12 are found to be, on average, noisier than readings coming from VPeN 11; this may be a symptom of an excessive wear of the sensors within the array.

Important features on single solutions. Results shown in figure 4.5 highlights that both solutions 5 and 7 can take advantage of feature selection (again, other solutions were too noisy to give meaningful results with the optimal settings suggested for random forest).

Let us evaluate which sensors are maximally discriminative for the conditioning factors, and



(a) Solution 5 (b) Solution 7

Figure 4.5: Features ranked according to their relevance for VPeN 12 single solutions

whether domain knowledge can give to these effects a proper explanation:

- for solution 5, the maximally discriminative sensor is MG 811;
- for solution 7, the two maximally discriminative sensors are MQ 137 and MG 811.

The analysis performed on IRSA Dataset in [95] show, for solution 7, the release of ammonia, which is directly related to the concentration of the solution. This may be due to anaerobic fermentation; furthermore, as described in [151], there are evidences that, at least in sediments, the presence of carbon dioxide is related to the presence of peptone. Therefore:

- results on solution 5 shows that the instrument is able to discern between different concentrations of a solution of soya peptone dissolved in a water matrix thanks to their different contributions in terms of carbon dioxide;
- results on solution 7 shows that the instrument is able to discern between different concentrations of a solution of milk poweder dissolved in a water matrix thanks to their different contributions in terms of both ammonia and carbon dioxide, due to anaerobic fermentation.

As for VPeN 11, this suggests that, with proper settings, the e-nose is capable to evaluate the impact of a conditioning parameter (in this case, concentration) on the overall solution.

In figure 4.6, the correlation analysis for the response of sensors MQ 137 and MG 811 on solution 7 is shown:

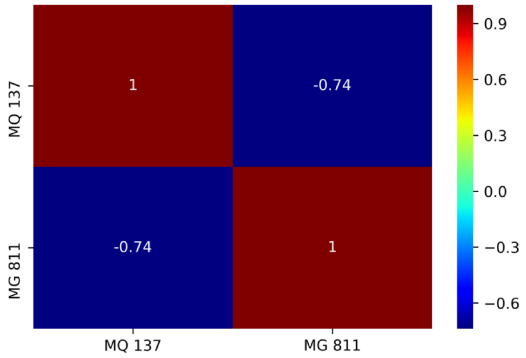


Figure 4.6: Correlation analysis through Kendall τ for solution 7 on VPeN 12

The response of both sensors appears to be anti-correlated; this, as explained in section 4.4.1, confirms that both the responses are relevant.

It may be interesting to analyze how correlation vary with the concentration of the solution within the water matrix. It appears that, with the lowest concentration, responses are correlated, with a value of $\tau_{low} = 0.63$. As the concentration increases, responses became slightly anti-correlated, with $\tau_{med} = -0.22$ and $\tau_{high} = -0.19$.

A possible interpretation of this effect lies in the fact that when the concentration of milk powder is low, effects of the release of either ammonia or carbon dioxide (or both) are negligible and, therefore, normalized responses for both sensors similarly depends on effects such as thermal noise. However, as the concentration of the solutions increases, the emissions of ammonia and carbon dioxide either increase or decrease over time, concealing the effects which were predominant when the concentration of the solution was low. Obviously, such hypothesis needs domain knowledge to be confirmed; however, it shows the possibility to discern between different signatures based simply on the evolution of the response of a set of sensors. This concept will be extended and applied to a real case in section 4.7.

Tables 4.12 and 4.13 show the clustering results using only the most meaningful features.

In this case, both scores benefits from the use of the subset of relevant features. The suggestion

Solution	ARI	ε	*min-pts*	Clusters	Exp. clusters	Outliers
5	0.87	0.23	12	4	3	0 %
7	1	0.47	12	3	3	0 %

Table 4.12: Best Adjusted Rand Index for VPeN 12 for most important features on single solutions of IRSA Dataset

that a proper setting, with a limited, yet *meaningful*, number of sensors, can easily outperform a complex, yet suboptimal, sensor array is therefore confirmed.

Solution	Silhouette	ε	*min-pts*	Clusters	Exp. clusters	Outliers
5	0.80	0.06	12	5	3	0 %
7	0.78	0.43	12	4	3	0 %

Table 4.13: Best Silhouette Score for VPeN 12 for most important features on single solutions of IRSA Dataset

Multiple solutions. In tables 4.14 and 4.15, results of the comparison of multiple solutions for VPeN 12 are shown.

Solutions	ARI	ε	*min-pts*	Clusters	Exp. clusters	Outliers
1-2	0.51	0.57	12	10	6	5 %
1-3	0.72	0.8	12	9	6	2.28 %
2-3	0.61	0.8	12	8	6	3.67 %
1-2-3	0.58	0.70	23	13	9	6 %
1-2-3-4	0.55	0.57	23	13	10	7.07 %
5-6-7-8	0.03	0.2	36	13	12	75.61 %

Table 4.14: Best Adjusted Rand Index for VPeN 12 on multiple solutions of IRSA Dataset

On average, results are aligned with the ones achieved by VPeN 11, with a remarkable difference found for the discrimination of solutions in the second tranche. As the instruments are identical, this suggests that one or more sensors from VPeN 12 suffer from either excessive use or poisoning.

Suggestions. It is clear that the results achieved for VPeN 11 are confirmed by VPeN 12, since, as already said, their settings are identical. However, some evidences suggest that VPeN 12 is somehow biased by an excessive use, or faulty hardware, as results are expected to be considerably better in several situations. This highlights how an exploratory data analysis can

Solutions	Silhouette	ε	*min-pts*	Clusters	Exp. clusters	Outliers
1-2	0.41	0.64	12	9	6	4.44 %
1-3	0.54	0.8	12	9	6	2.28 %
2-3	0.51	0. 68	12	9	6	4.5 %
1-2-3	0.47	0.04	27.65	1	9	98.03 %
1-2-3-4	0.38	0.8	12	9	10	2.37 %
5-6-7-8	0	0.08	31	1	12	98.86 %

Table 4.15: Best Adjusted Rand Index for VPeN 12 on multiple solutions of IRSA Dataset

be used not only to discover underlying data processes, but also how it can be exploited to find inconsistencies within the *producer* of these data, therefore directing further investigation aimed at improving the acquisition settings. Let us now evaluate the last part of the analysis on the IRSA Dataset, which concerns the comparison of data acquired by both VPeNs.

Comparison between VPeNs

As already stated, the two VPeNs used in the comparison are supposed to be identical, as they perform the same experiments under the same settings. One should therefore expect that data coming from the two instruments, once normalized, reflect the the same underlying process. As a consequence, one of the main assumptions on which this set of experiments has been based is that the number of expected clusters should *exactly the same* as shown for VPeNs 11 and 12. For the rest, the protocol that has been used is the same as the one used for the experiments on single instruments.

Single solutions. In table 4.16, the comparison between the results given by the two VPeNs on single solutions is shown.

Interestingly, results perfectly reflect what has been shown for single VPeNs: on average, the ARI is low, while a negligible number of outliers is found. As this evaluation takes into account also the *repeatability* of the measures, for the aforementioned reasons, it is likely that the effects which have been described for VPeN 12 (that is, a probable excessive usage or poisoning of sensors within the array), along with the suboptimal design of the measurement chamber, can be identified as the causes of such poor performance.

Solution	ARI	ε	*min-pts*	Clusters	Exp. clusters	Outliers
1	0.37	0.47	13	7	3	2.84 %
2	0.2	0.2	12	13	3	8.44 %
3	0.48	0.55	12	9	3	1.27 %
5	0.52	0.8	12	7	3	1 %
6	0.5	0.8	12	8	3	0.39 %
7	0.56	0.8	12	6	3	1.45 %
8	0.36	0.70	12	5	3	0 %
A	0.36	0.53	29	7	3	1.83 %

Table 4.16: Best Adjusted Rand Index in the comparison of single solutions acquired by both VPeNs.

In table 4.17, results for the silhouette score on data coming from both VPeNs substantially confirm the achieved results.

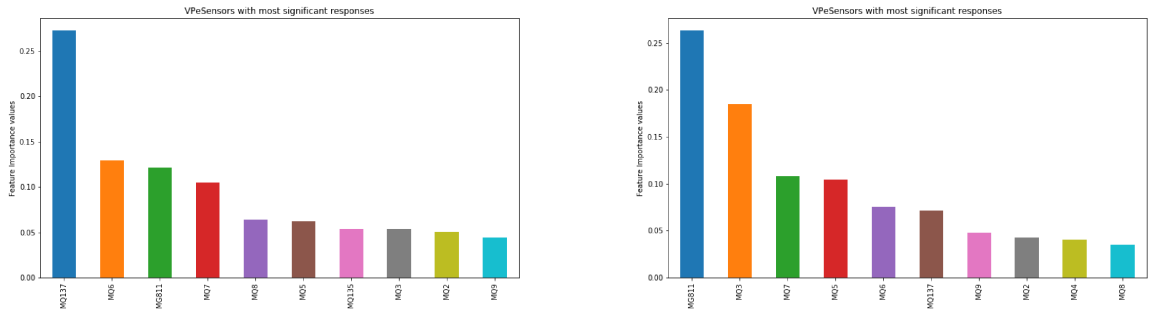
Solution	Silhouette	ε	*min-pts*	Clusters	Exp. clusters	Outliers
1	0.83	0.72	12	3	3	0.67 %
2	0.67	0.76	15	2	3	2.39 %
3	0.61	0.49	12	10	3	2.05 %
4	0.73	0.78	28	2	1	10.33 %
5	0.53	0.74	31	11	3	5.67 %
6	0.55	0.53	31	10	3	4.72 %
7	0.57	0.72	12	6	3	1.5 %
8	0.69	0.70	12	5	3	0 %
A	0.75	0.78	16	4	3	0.28 %

Table 4.17: Best Silhouette Score in the comparison of single solutions acquired by both VPeNs.

Important features on single solutions. In figure 4.7, feature ranking for solutions 7 and a, according to a random forest classifier, are shown. Results show that:

- for solution 7, the most discriminative sensor is MQ 137;
- for solution A, the most discriminative sensors are MG 811 and MQ 3.

In figure 4.8, the correlation for the most meaningful features for solution a is shown. Following the same considerations made in previous scenarios, it is clear to see how the response of MG 811 and MQ3 are anti-correlated.



(a) Solution 7 (b) Solution a

Figure 4.7: Features ranked according to their relevance for combined VPeNs - single solutions

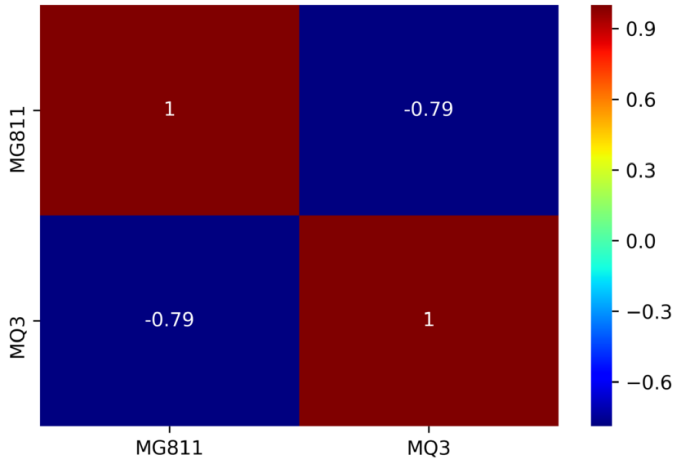


Figure 4.8: Correlation analysis between the responses of sensors MG 811 and MQ 3 or results from both VPeNs on solution A.

Results show that the overall response from these sensors are anti-correlated. An per-concentration analysis highlights that, for the lowest possible concentration value, sensors have a negative correlation ($\tau_{low} = -0.15$), for the intermediate concentration responses are almost uncorrelated ($\tau_{int} = 0.02$), while for the highest concentration responses are slightly negatively correlated ($\tau_{high} = -0.26$).

A chemical interpretation of the results achieved on solution A is not given in [95], therefore further domain knowledge would be needed to characterize the fact that one of the most discriminating responses is coming from sensor MQ 3, which, from table 4.3, is able to sense for alcohol and gasoline. As for the milk powder, the most discriminative response given by MQ

137 may be related to the fermentation caused by bacteria within the solution.

In table 4.18 the best ARI for solutions 7 and A on most relevant features is reported. Interestingly, both results are lower than the results achievable when using the full feature set.

Solution	ARI	ε	*min-pts*	Clusters	Exp. clusters	Outliers
7	0.5	0.45	12	4	3	0 %
A	0.18	0.02	35	5	3	8.67 %

Table 4.18: Results of supervised DBSCAN for combined VPeNs on IRSA dataset - multiple solutions

As for silhouette scores, as shown in table 4.19, it is almost equal to one in both cases, therefore clusters are well defined in the feature space.

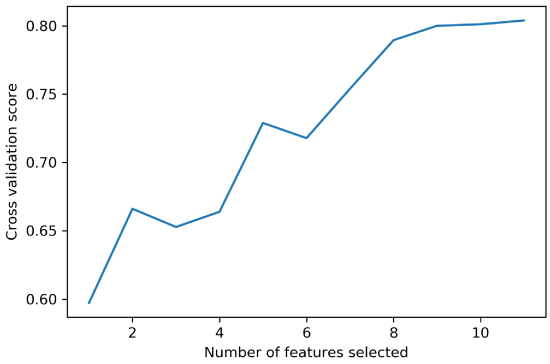
Solution	Silhouette	ε	Samples	Clusters	Exp. clusters	Outliers
7	0.98	0.12	12	7	3	0.72 %
A	0.91	0.16	12	2	3	0 %

Table 4.19: Results of unsupervised DBSCAN for combined VPeNs on IRSA dataset - multiple solutions

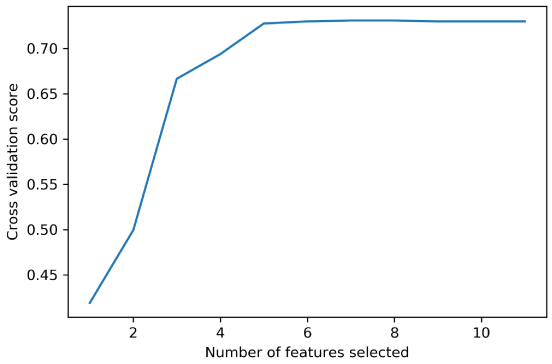
One may suggest to lower the threshold for feature relevance, as already done for VPeN 11, to include a wider (and, possibly, more comprehensive) set of features. However, results do not improve even when the relevance threshold is set to 0.10, with a ARI for solution 7 of 0.24, and of 0.35 for solution A.

The optimal number of important features can be evaluated through recursive feature elimination (RFE), which recursively prune the least important features from the feature set according to a performance score of an estimator [158]. This procedure can be enhanced using cross validation to reinforce achieved results.

Figure 4.9 shows results of such evaluation. It is possible to see that the optimal number of suggested features is 11 (for solution 7) and 8 (for solution A). Hence, in these specific cases, it is suggested to run the experiments without performing feature selection first, due to the fact that the optimal number of features is almost equal to the actual number of features in the dataset.



(a) Solution 7



(b) Solution a

Figure 4.9: Results of RFE with ten rounds of cross-validation on solutions 7 and A.

Solutions	ARI	ε	*min-pts*	Clusters	Exp. clusters	Outliers
1-2	0.36	0.37	29	13	6	10.67 %
1-3	0.44	0.49	35	11	6	5.81 %
2-3	0.42	0.74	18	6	6	0.89 %
1-2-3	0.33	0.51	35	13	9	5.09 %
1-2-3-4	0.40	0.41	30.79	13	10	4.03 %
5-6-7-8	0.37	0.78	36	13	12	1.29 %

Table 4.20: Best adjusted rand score for combined VPeNs on multiple solutions

Multiple solutions. In tables 4.18 and 4.21, results for adjusted rand index and silhouette score are shown on the selected combination of substances.

Again, it appears that labels assigned by the best clustering in terms of ARI do not properfly fit the ground truth, even if the number of outliers found within data is, on average, low.

As for silhouette score, results highlight that, on average, when the selected solutions are compared, clusters which take form are sufficiently separated in the feature space, with a low number of outliers. This does not hold for solutions 5, 6, 7 and 8, where, even if the number of

Solutions	Silhouette	ε	*min-pts*	Clusters	Exp. clusters	Outliers
1-2	0.84	0.8	20	3	6	0.67 %
1-3	0.66	0.70	16	7	6	0.44 %
2-3	0.67	0.61	12	9	6	0.89 %
1-2-3	0.65	0.7	16	8	9	0.48 %
1-2-3-4	0.71	0.57	12	8	10	0.55 %
5-6-7-8	0.24	0.78	36	13	12	1.29 %

Table 4.21: Best silhouette score for combined VPeNs on multiple solutions

clusters which is found is almost correct, these are found to be highly overlapped.

4.4.3 Results on ISMAR Dataset

In this section, results achieved on the ISMAR Dataset are shown. The experimental protocol which has been followed is the same used in the experiments on the IRSA Dataset.

VPeN 11

Single solutions. Table 4.22 shows the best ARI on single solutions of ISMAR Dataset.

Solution	ARI	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.26	0.8	12	13	3	20.45 %
C	0.69	0.8	12	6	3	6 %
D	0.80	0.8	12	5	3	3.78 %

Table 4.22: Results of supervised DBSCAN for VPeN 11 on ISMAR dataset - single solution

In this case, the value for the ARI is low for the solution B, which also shows several outliers; however, it is considerably higher on solutions C and D. As for the silhouette scores, which are shown in table 4.23, results show that a clear separation between clusters cannot be achieved. It is important to note that the best performance in terms of both ARI and silhouette score are achieved with exactly the same configuration for solutions B and C (this does not hold for solution D).

Solution	Silhouette	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.21	0.8	12	13	3	20.45 %
C	0.48	0.8	12	6	3	6 %
D	0.37	0.78	17	4	3	3.78 %

Table 4.23: Results of unsupervised DBSCAN for VPeN 11 on ISMAR dataset - single solution

Interestingly, these results are confirmed by [98], where it has been shown, through a PERMANOVA analysis, that data do not allow to highlight the effect of different heating temperatures.

Important features on single solutions. In table 4.24, results achieved while selecting only the most important features on solutions B and D are shown.

Solution	ARI	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.78	0.37	12	4	3	0.22 %
D	1	0.08	12	3	3	0 %

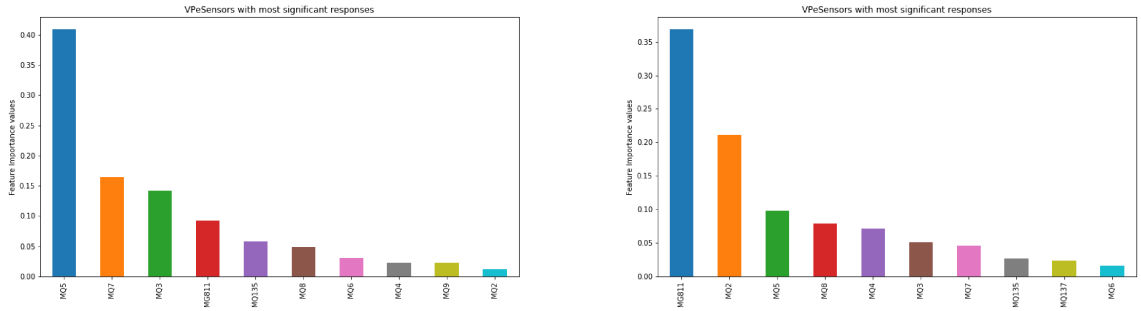
Table 4.24: Results of unsupervised DBSCAN for VPeN 11 with most important features selected on ISMAR dataset - single solution

Results show a considerably better improved value for ARI; this indicates that, in this case, feature selection properly works. These results are confirmed by the improvements achieved by the silhouette score.

Solution	Silhouette	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.98	0.37	12	4	3	0.22 %
D	0.85	0.18	12	2	3	0 %

Table 4.25: Results of unsupervised DBSCAN for VPeN 11 with most important features selected on ISMAR dataset - single solution

In figure 4.10, feature ranking resulting from the application of a random forest classifier on solutions B and D is shown.



(a) Solution b

(b) Solution d

Figure 4.10: Features ranked according to their relevance for VPeN 11 on ISMAR dataset

It can be seen that responses which have been considered are:

- for solution B, the response from sensor MQ 5;
- for solution D, the response from sensor MG 811.

Let us recall that sensor MQ 5 can sense either LPG, hydrogen of methane, while sensor MG 811 can sense carbon dioxide. Solution B is given by water taken from a tank for the production of mussels before the insertion of the bivalves; however, the responsiveness of sensor MQ 5 to different values of temperature, considering also the presence of highly volatile gases (such as methane and LPG) suggests that water is not perfectly 'clear', but there are instead residuals from previous cycles of bivalves aquaculture (and, therefore, residuals from mussels digestion, such as methane [99]. As for solution D, further chemical analysis may be needed to evaluate the differences in the release of carbon dioxide as temperature varies.

Multiple solutions. In table 4.26, the adjusted rand index for the comparison of solutions b, c and d are shown. In this case, VPeN 11 does not perform well in discriminating between different substances.

Solutions	ARI	ε	Samples	Clusters	Exp. clusters	Outliers
B-C-D	0.59	0.66	12	12	9	4.19 %

Table 4.26: Results of supervised DBSCAN for VPeN 11 on ISMAR dataset -multiple solutions

Also the silhouette score, as shown in 4.27, is low, therefore clusters are overlapped.

Solution	Silhouette	ε	Samples	Clusters	Exp. clusters	Outliers
B-C-D	0.50	0.8	12	7	9	2.26 %

Table 4.27: Results of unsupervised DBSCAN for VPeN 11 on ISMAR dataset -multiple solutions

By comparing these results with the ones achieved by Cilenti in [98], it is clear that the VPeN can correctly discriminate between an alert (that is, the presence of the compounds within solution D) and a normal situation (supposedly normal water, such as solutions B and C). However, when all these solutions are compared, VPeN does not appear to be able to correctly discriminate between solutions as temperature changes.

VPeN 12

Single solutions. Let us evaluate the results achieved by the second VPeN. In table 4.28, the best adjusted rand index on each solution belonging to ISMAR dataset is shown. It is important

to note that solution D envisages only for two clusters as only data for two temperatures (30 C and 45 C) were retrieved from data repository.

Solution	ARI	ε	*min-pts*	Clusters	Exp. clusters	Outliers
B	0.31	0.8	12	10	3	20.67 %
C	0.44	0.8	12	11	3	16.78 %
D	0.66	0.8	12	3	2	9 %

Table 4.28: Best adjusted rand index for VPeN 12 on single solutions for ISMAR dataset

Experiments show how the achievable best ARI is relatively low (even for solution D, where only two clusters are expected). This is confirmed also by the poor results achieved in terms of silhouette score, as shown in table 4.29.

Solution	Silhouette	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.31	0.8	13	8	3	26.11 %
C	0.30	0.8	14	5	3	28.56 %
D	0.67	0.78	12	3	2	9.33 %

Table 4.29: Best silhouette score for VPeN 12 on single solutions for ISMAR dataset

Important features on single solutions. However, as for the VPeN 11, feature selection can greatly improve clustering performance. Let us first evaluate table 4.30, which shows the best adjusted rand index for the most meaningful set of features for solutions B, C and D.

Solution	ARI	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.74	0.20	26	4	3	0.67 %
C	0.42	0.10	27	4	3	0 %
D	1	0.60	12	2	2	0 %

Table 4.30: Best adjusted rand index for most important features for VPeN 12 on single solutions for ISMAR dataset

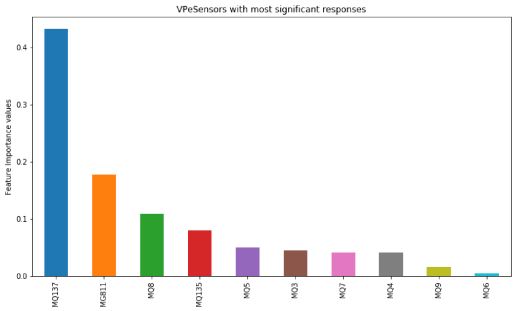
If compared to table 4.28, it is clear how, except for solution C, feature selection slightly improves clustering performance, and, for solution D, DBSCAN is capable to achieve the best possible scoring.

The same considerations hold for table 4.31, which shows a great improvement with respect to the case in which all features are used for clustering.

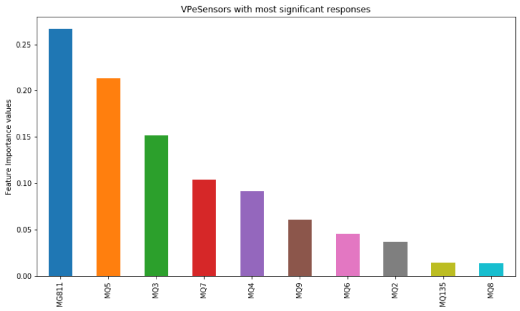
Solution	Silhouette	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.96	0.20	26	4	3	0.67 %
C	0.66	0.08	13	7	3	1.67 % %
D	0.82	0.59	12	2	2	0 %

Table 4.31: Best silhouette score for most important features for VPeN 12 on single solutions for ISMAR dataset

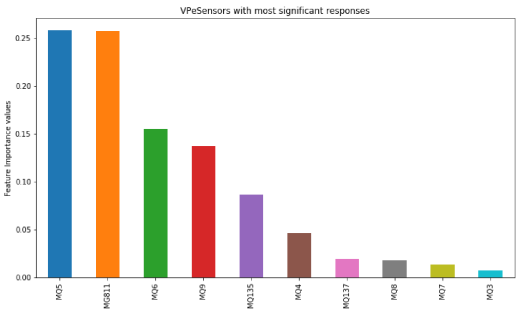
If the relevance threshold is lowered to 20 % for solution C, the best ARI which could be achieved is of 0.67, while the best silhouette score is of 0.65. In this case, two sensors are considered, that is, sensor MG 811 and MQ 5. The correlation analysis for the response from these two sensors is shown in figure 4.12.



(a) Solution b



(b) Solution c



(c) Solution d

Figure 4.11: Features ranked according to their relevance for VPeN 12 on ISMAR dataset

It is interesting to note that data appear to be correlated. It can be noted that this correlation is due to effects which manifest themselves at the temperatures of 30 C and 60 C, as at 45 C values are anti-correlated. This behavior should be properly characterized using domain

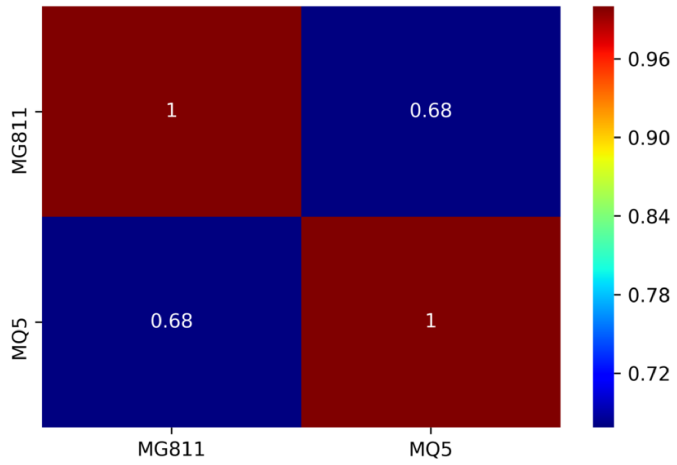


Figure 4.12: Correlation analysis for solution C on ISMAR Dataset.

knowledge.

Multiple solutions. As for the comparison of multiple solutions, results are shown in tables 4.32 and 4.33.

Solutions	ARI	ε	Samples	Clusters	Exp. clusters	Outliers
B-C-D	0.62	0.41	12	13	9	5.62 %

Table 4.32: Best adjusted rand score for VPeN 12 on multiple solutions for ISMAR dataset

Results essentially confirm what has been achieved by VPeN 11, even if, in this case, it appears that VPeN 12 achieved slightly better performance if compared to the other instrument.

Solutions	Silhouette	ε	Samples	Clusters	Exp. clusters	Outliers
b-c-d	0. 70	0.78	13	5	9	3.58 %

Table 4.33: Best silhouette score for VPeN 12 on multiple solutions for ISMAR dataset

VPeN Combined

Single solutions. Finally, let us show results achieved on single solutions of the ISMAR dataset when using data acquired from both the VPeNs.

In table 4.34, it is shown as values for ARI is considerably low on each one of the solution. Silhouette score (table 4.35) are slightly better.

Solution	ARI	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.21	0. 27	25	9	3	21.94 %
C	0.36	0.33	32	9	3	17 %
D	0.41	0.64	12	6	2	3.08 %

Table 4.34: Results of unsupervised DBSCAN for combined VPeNs on ISMAR dataset - single solution

Solution	Silhouette	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.62	0.76	14	4	3	1.33 %
C	0.70	0.74	14	4	3	2.11 %
D	0.70	0.74	16	4	2	4.42 %

Table 4.35: Results of unsupervised DBSCAN for combined VPeNs on ISMAR dataset - single solution

Important features on single solutions. Also in this case, when only important features are used for clustering, results do not improve for solutions B and D. This is probably due to the effects described for VPeN 12.

Silhouette score, however, is greatly improved, as shown by results reported in table 4.37.

By lowering threshold to 20 % for solution B, also the response of sensor MQ 5 can be considered, slightly improving the best achievable ARI (to a value of 0.34) while worsening the best achievable silhouette score (to a value of 0.79). This effect can also be seen for solution D, as lowering the detection threshold to 15 % (lowering it to a higher value would mean to not consider any other feature) improves results in terms of best achievable ARI (0.48) while worsening the best achievable silhouette score (0.80). Hence, in this case, feature selection may not be improve overall results.

Multiple solutions. As for results achievable on multiple solutions, these are reported in the following tables.

Solution	ARI	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.17	0.25	26	4	3	0.33 %
D	0.45	0.02	12	5	2	1.5 %

Table 4.36: Results of unsupervised DBSCAN for combined VPeNs with most important features selected on ISMAR dataset - single solution

Solution	Silhouette	ε	Samples	Clusters	Exp. clusters	Outliers
B	0.98	0.25	26	4	3	0.33 %
D	0.90	0.06	12	3	2	0 %

Table 4.37: Results of unsupervised DBSCAN for combined VPeNs with most important features selected on ISMAR dataset - single solution

Solution	ARI	ε	Samples	Clusters	Exp. clusters	Outliers
B-C-D	0.41	0.43	13	13	8	3.17 %

Table 4.38: Results of supervised DBSCAN for combined VPeNs on ISMAR dataset -multiple solutions

It is clear that both the best achievable ARI and the best achievable silhouette score are considerably low.

Solution	Silhouette	ε	Samples	Clusters	Exp. clusters	Outliers
B-C-D	0.62	0.70	12	11	8	0.96 %

Table 4.39: Results of unsupervised DBSCAN for combined VPeNs on ISMAR dataset -multiple solutions

4.4.4 Discussion

Results depicted in this section highlight a complex situation, due to an ill-conditioned experimental design: in fact, by simply comparing substances defined in [95] and [98] with the sensors in the measurement chamber of the VPeNs, one may expect that many of the parameters which are considered relevant for the analysis cannot be directly found by the actual settings of the sensor.

However, as these data were acquired during a prototypical stage test, they should be used, along with the proposed techniques, to refine both the design and the acquisition methodologies. In fact:

- it has been proved that the VPeNs are able to respond to several conditioning parameters, given a proper knowledge of the domain, *which should always lead the selection of sensors within the measurement chamber*;

- it has been shown as feature selection can greatly improve achievable results, with the exception of some specific cases, which should be investigate using domain knowledge;

- it has been suggested that the biases due to the environmental settings can be addressed in a conditioning step, which should consider the external environmental conditions; such method should be one of the first upgrades applied to the instrument;

- it has also been proved that an extensive calibration step is needed, as there are many (and often hardly modelable without enough data) biases introduced by factors such as different usages of the sensors within the VPeN, or even different working temperatures or turbulence conditions.

4.5 Classification with Deep Neural Networks

The goal of the second set of experiments performed on the VPeN Datasets was to evaluate the use of machine learning to classify data acquired by the instrument.

To this end, a deep artificial neural network (ANN) [22] has been used. This network is built on the concept of *multilayer perceptron* (MLP), introduced by Rosenblatt in 1958 [23], which is considered as the 'foundation' for deep learning.

For the selected network, an architecture composed by three (hidden) fully connected layers whose activation function is a rectified linear unit (ReLU) [13]. The use of this activation function is suggested as Krizhevsky demonstrated, with AlexNet, that it can achieve the same results that can be obtained with other activation functions, while substantially lowering the computational cost needed to execute the network. In the last layer (the output layer), a classical dense, fully connected layer has been used, with a softmax activation function to perform classification [155]. As for the loss function, categorical cross-entropy has been used [155]. To train the network, the Adam optimization algorithm has been used [156]. In the training phase, a k-fold cross validation procedure, with $k = 10$, has been used to validate results achieved by the network. The use of dropout layers [157] between hidden fully connected

layers has also been evaluated.

Results evaluation

First, it is important to underline that VPeN Datasets are *imbalanced*, meaning that the quantity of data belonging to each experiment may differ according to the number of measurement/cleaning cycles used (usually, two or three). If such imbalance does not afflict an agglomerative clustering method as DBSCAN, it does have a negative impact on a learning algorithm such as a deep neural network. In these situation, usually, three approaches can be followed [155]:

- fine-tune the initial weights, to support specific classes of data;
- randomly remove samples of most represented classes (hard negative mining);
- accept the imbalance.

The third approach is usually the one to prefer when enough quantities of data are available, and it has been shown that good performance can be achieved [118]. In these trials, the imbalance has been accepted, as data have been proven themselves to be enough to deal with the number of parameters (weights and biases of neurons) of the network.

Let us show the first batch of results, which have been performed on non-normalized data. In this case, after 15 epochs of training for each one of the 10 validations, the model achieves a mean accuracy of 82.2 %. In figure 4.13, the confusion matrix for this experiment is shown.

The confusion matrix shows how the highest number of misses are achieved when trying to classify solution 6.

The achieved performance are overall good; however, they could be dramatically improved by using some of the hints that came from the analysis performed in sections 4.4.2 and 4.4.3, that is, data appear to be afflicted by several biases due to systemic errors and offsets within the readings of the sensors. Therefore, by just applying a normalization procedure (that is, data are

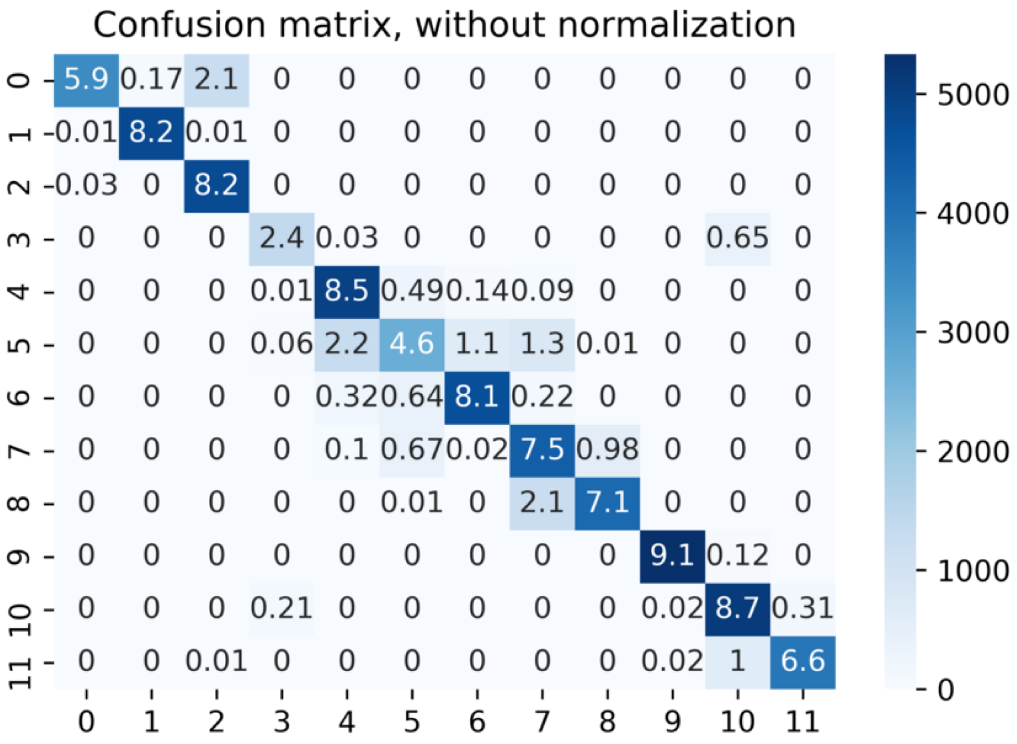


Figure 4.13: Confusion matrix with data without normalization

scaled to assume a normal distribution with zero mean and unitary variance), it is possible to achieve an average accuracy on the 10-validation procedure of 98.16 %. The confusion matrix (which is, obviously, almost diagonal) relative to this case is shown in figure 4.14.

As for the dropout, the effects of such layers have been testing imposing a dropout rate (that is, the percentage of the total neurons which are not considered at each training iteration) from 10 % to 50 %, which is the value which has been found to guarantee the best performance in case of overfitting [119]. However, in this case, cross-validation accuracy was reduced to 96.45 % when the dropout rate was of 10 %, and to 58.65 % when the dropout rate was of 50 %. Hence, it can be concluded that no dropout layers are needed, and the selected configuration for the network does not suffer from overfitting.

Now, let us recall the concept which lead the comparison between the results of the two VPeNs, as shown in sections 4.4.2 and 4.4.3, that is: once data coming from two identical sensors, exposed to identical experimental settings, are normalized, they should resemble the same

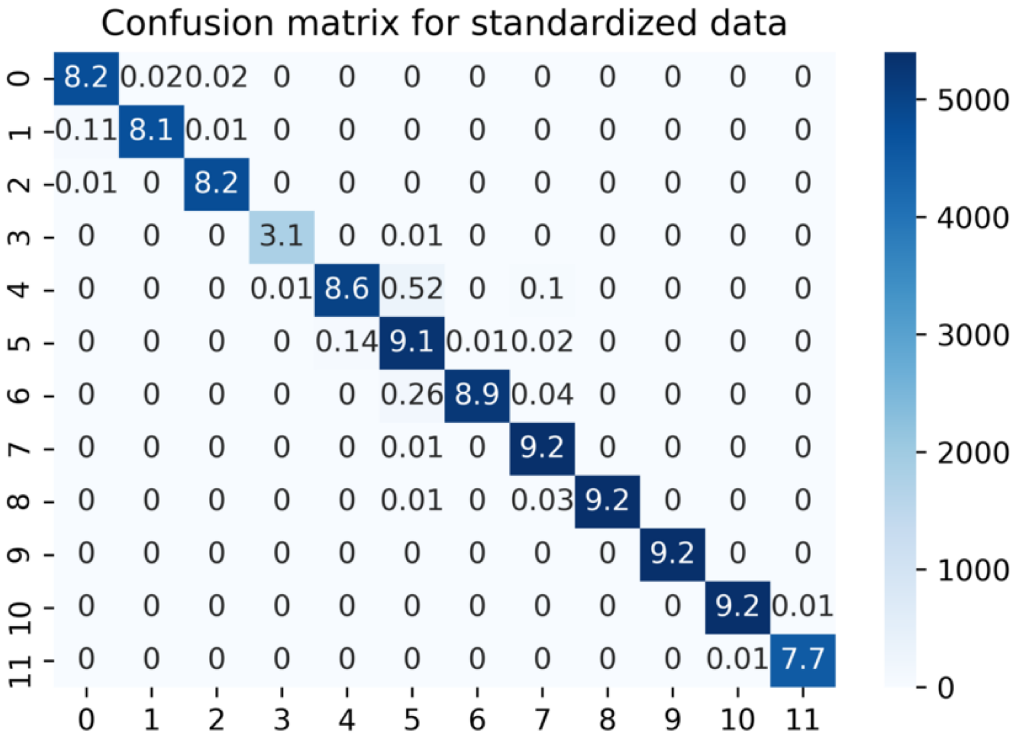


Figure 4.14: Confusion matrix with data with normalization

data generation process. Therefore, if this condition is met, a network trained exclusively on data coming from VPeN 11 should have comparable accuracy on data coming from VPeN 12, and vice versa. As already pointed out, several biases do not allow to achieve this kind of repeatability on the VPeN Dataset; however, here, a method to overcome this problem, and allow to a network trained on a VPeN to be effective also on data acquired by the other without a complete retraining, is shown.

Transfer learning

Transfer learning is an idea introduced by Szegedy in [9], and is based on the concept of *feature abstraction* in a deep neural network. Let us briefly consider a deep convolutional neural network for image recognition. Within this network, first layers are used to model generic shapes, such as corners or edges, while later layers represent more complex layers, often strictly related to the dataset on which the network is being trained.

Intuitively, this idea can be borrowed, and applied to this specific case, where complex biases exists between (supposedly) identical instruments. Specifically, two networks, with the same configuration of the network used in the most generic case, have been trained from scratch on data coming from VPeN 11 and VPeN 12, achieving an accuracy of 99.27 % and 97.69 %, respectively.

These networks, as expected, achieved poor results on the other instrument: as for the network originally trained on data from VPeN 11, it achieved an accuracy of only 15.95 % on data from VPeN 12, while the network trained on VPeN 12 achieved an accuracy of 22.91 % on data coming from VPeN 11.

By applying transfer learning, retraining on the new dataset only the last two hidden layers (and, obviously, the output layer), results have significantly improved: as for the network originally trained on VPeN 11, it could achieve an accuracy of 87,69 % on VPeN 12 with transfer learning, and similar results have been achieved by the network originally trained on VPeN 12, with an accuracy of 82.38 % on VPeN 11.

The interpretation of such results is straightforward: first layers of the network capture generic, and common, behaviors of the instruments, while later layers capture phenomena characteristics of the specific instrument, and which actually allow to model the existing biases between the e-noses.

4.6 Results on IRSA - Wastewater

In this section, results of univariate modeling of a subset of the compounds found within the IRSA Wastewater Dataset are shown.

The selection of the compounds on which the analysis has been performed followed the direction given in[95], which highlights how only data gathered from the wastewater treatment plant of Vimercate should be considered, as more relevant from a biochemical perspective. Furthermore, to follow one of the main assumption on which Box and Jenkins based the analysis of temporal

series, only compounds with more than 50 samples available have been considered for the analysis.

Hence, once the relevant compounds have been found, the following procedure has been used for the analysis.

- An exploratory analysis is performed on each compound, to gather knowledge on the specific time series. Specifically, as already shown in chapter 2, the ACF and PACF functions, along with the normal Q-Q plot and the histogram, have been explored. Furthermore, the results of an STL decomposition are analyzed to roughly evaluate the presence of trends and/or seasonal effects.

- After the exploratory analysis, the optimal seasonal ARIMA model has been found for each compound. To this end, a grid search on two triples of hyper-parameters, which represent the orders (p, d, q) and (P, D, Q) (cfr. chapter 2), is performed, with the specific goal to minimize both the Akaike Information Criterion and the mean squared error between the ground truth (a validation set which represents the last six months of the series itself) and the found model.

- The residuals of the achieved SARIMA model are evaluated, to ensure that are uncorrelated and normally distributed, with zero mean and unitary variance [19].

It is important to underline that time series are not densely or regularly sampled. Therefore, experiments have been performed on both the original time series, and their resampled versions, where missing samples were taken on a daily basis, using a cubic spline interpolation. This approach was suggested by similar works in fields such as astronomy [108] and genetics [107].

4.6.1 Nitrogen Compounds

Chemical consideration. The first compounds which have been analyzed are the ones related to the total nitrogen. These are found within the wastewater in three different forms,

that is, *ammonia*, *nitric oxide* and *nitrous oxide*. As shown in [95], each of these parameter show high variability over time, and both ammonia and nitrous oxide are often above the allowed thresholds, which are of 30 mg/l for ammonia, and 0.6 mg/l for nitrous oxide. As for the nitric oxide, the parameter is considered, from the chemical perspective, negligible, as it is often below the detection threshold. In the following, each one of these three forms will be analyzed separately.

Ammonia

Time series Exploratory Data Analysis. The time series relative to ammonia is shown in figure 4.15. By observing the time series itself, it appears to be randomly distributed around an average value of about 35 mg/l. However, an analysis of both the ACF and the PACF plots shows that, according to the given made in chapter 2, the series shows a slowly decaying ACF, while the PACF is cut off after two lags, which may be an indication of an outgoing AR process. As for the normal Q-Q plot and the histogram, they resemble a normal behavior, even if the histogram appears to be slightly skewed.

As already said, the time series has not been sampled on regular basis. As a consequence, in the (additive) STL decomposition, shown in figure 4.16, the sampling period has been set to 3 days, according to the number of samples available within the first year of the series (almost 120 samples), therefore the seasonality effect has been supposed to be exhibited every 120 samples. This procedure has been performed on each of the parameters which will be shown in the following, obviously fixing the seasonality according to the number of samples within the time series.

The STL decomposition shows an overall stable trend within data. Residuals appears to be randomly distributed, and a slight seasonal effect is reported.

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
1	1	9	0	0	0	2045.07	123.19

Table 4.40: Parameters found for best seasonal ARIMA model on ammonia time series.

SARIMA modeling. Given the aforementioned results, let us evaluate the parameters found by the SARIMA model.

Interestingly, there is an AR contribution with $p = 1$, and an MA contribution with $q = 9$, which is almost exactly where the ACF plot cuts off. Furthermore, achieved results do not envisage for seasonal effects; therefore, the (superimposed) seasonal effects shown by the STL decomposition are not found to be relevant by the SARIMA model. It must be underlined that this could be expected from the exploratory analysis of the series itself. Finally, the mean squared error on the last six months is relevant; as it can be seen from figure 4.17, the SARIMA model is partially able to follow the original time series, but has several issues modeling quick spikes.

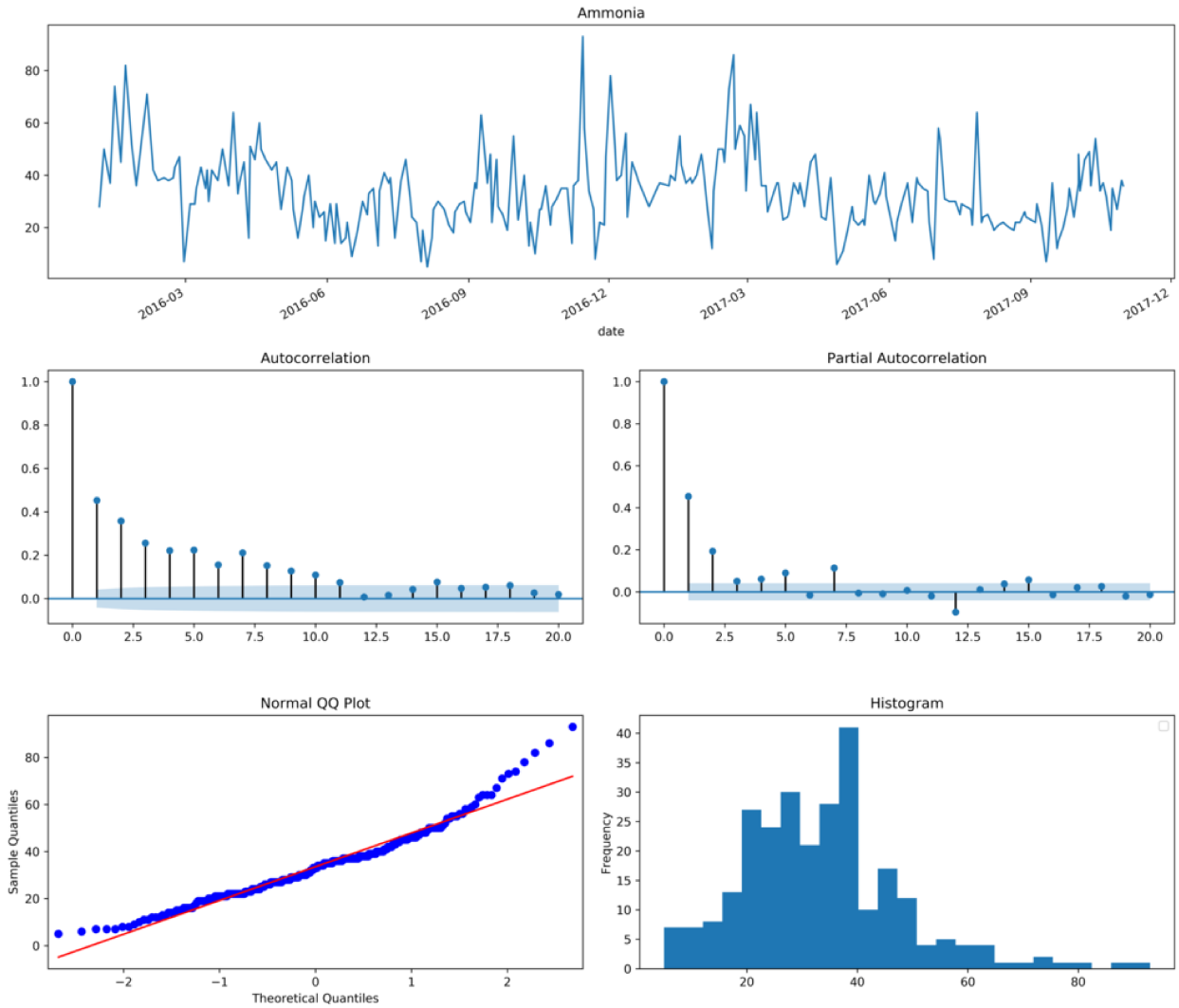


Figure 4.15: Analysis of ammonia for Vimercate Wastewater Treatment Plant

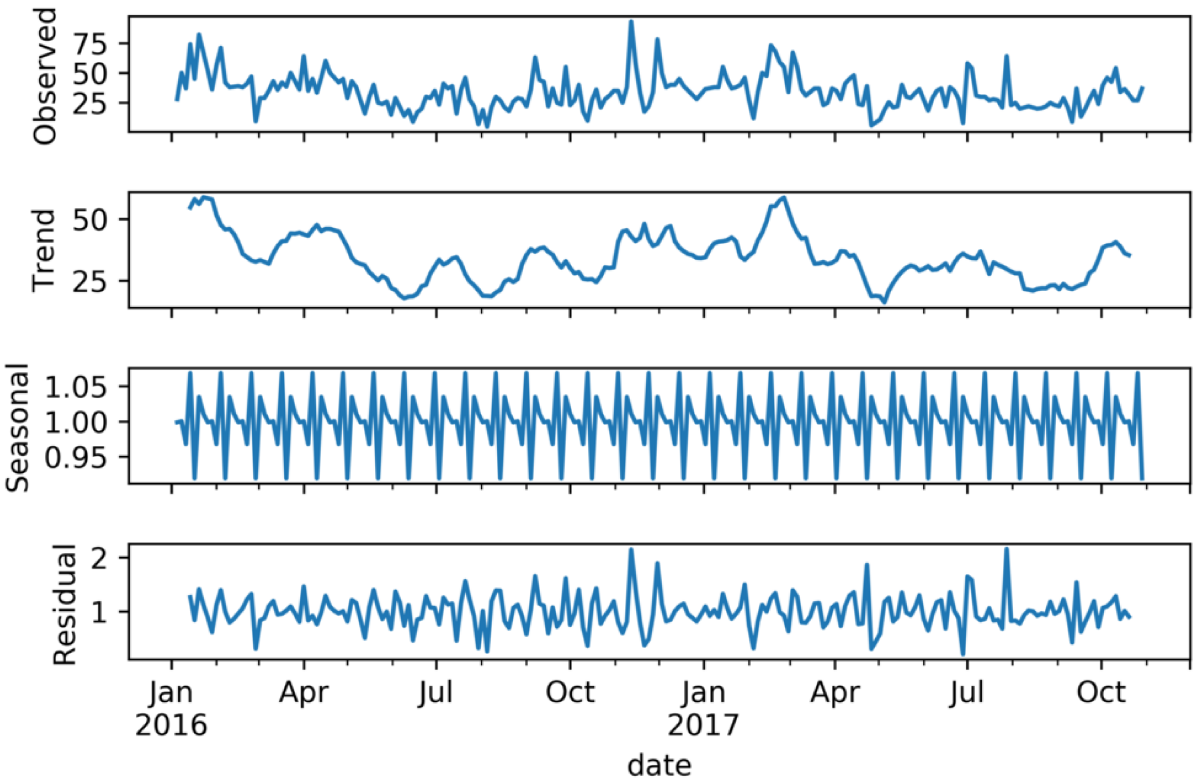


Figure 4.16: STL decomposition for ammonia

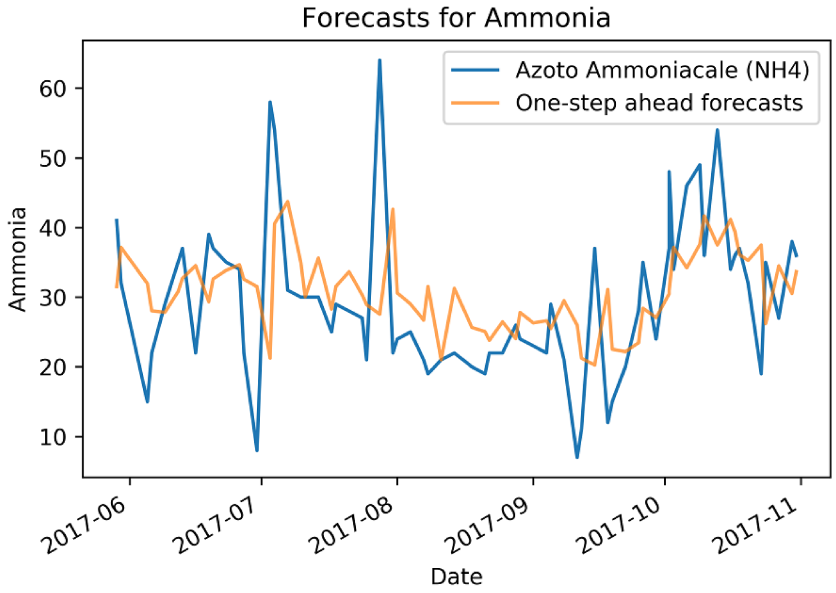


Figure 4.17: Forecasts for best seasonal ARIMA model on ammonia time series.

Analysis can be further extended using the diagnostics shown in figure 4.18:

- from the histogram, it can be seen that the KDE plot resemble $N(0,1)$, which is a normal distribution with zero mean and standard deviation equals to 1, therefore, residuals are normally distributed;
- from the normal Q-Q plot, the ordered distribution of residuals mostly follows the linear trend of the samples taken from the normal distribution $N(0,1)$. A slight deviation, which can be also seen observing the kernel density estimation plot, can be found for higher values;
- from the standardized residuals plot, it is clear that no obvious seasonal effects can be found within residuals. This is also confirmed by the correlogram, which shows low correlation between lagged residuals.

Achieved results, however, are unsatisfactory in terms of MSE on prediction. Therefore, it is important to evaluate results achievable by re-sampling the time series through cubic spline interpolation.

SARIMA modeling on resampled time series. Let us evaluate the parameters found for the resampled time series. In this case, the period of seasonality is set to 365 (this procedure will be extended to all the following cases).

AR	I	MA	sAR	sI	sMA	AIC	MSE forecasting
4	1	9	0	0	0	4203.5	20.13

Table 4.41: Parameters found for best seasonal ARIMA model on resampled ammonia time series.

Again, no seasonal effects are highlighted by the best model. Interestingly, the AIC is higher than the one found for the original time series. However, the AIC can be used to compare models obtained on the same dataset, therefore this value cannot be compared with the one achieved in the previous evaluation, but only to compare different SARIMA models achieved on the same data.

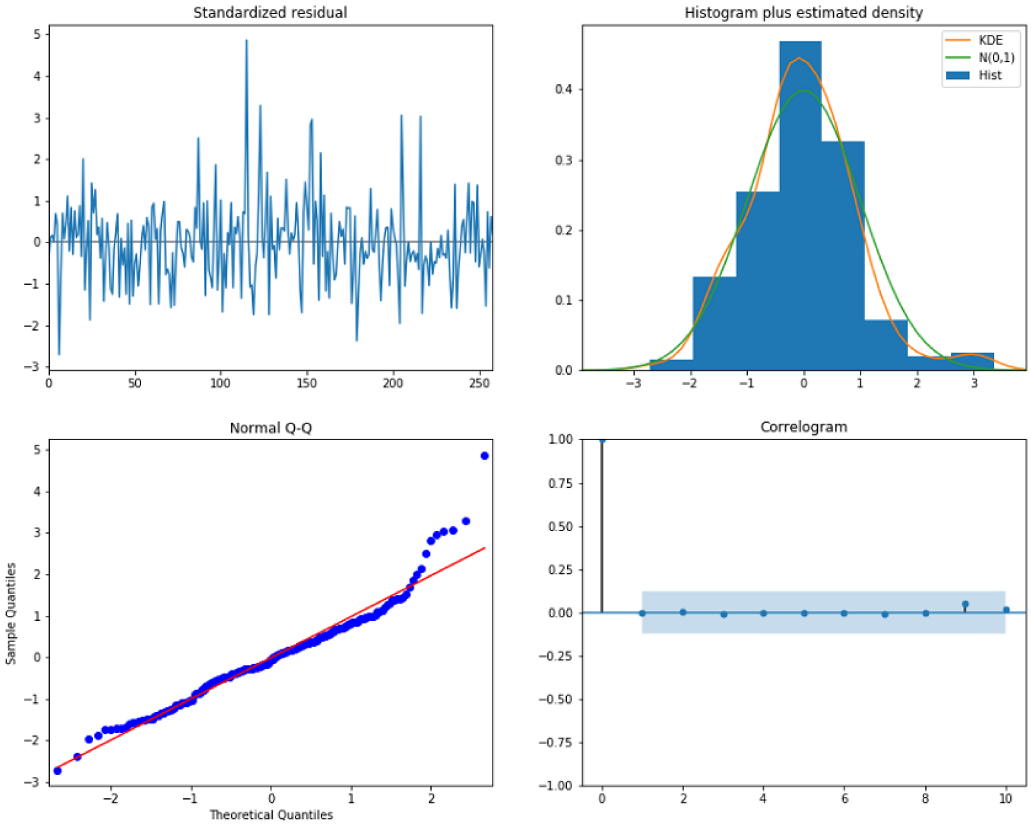


Figure 4.18: Diagnostics for best SARIMA model on ammonia time series.

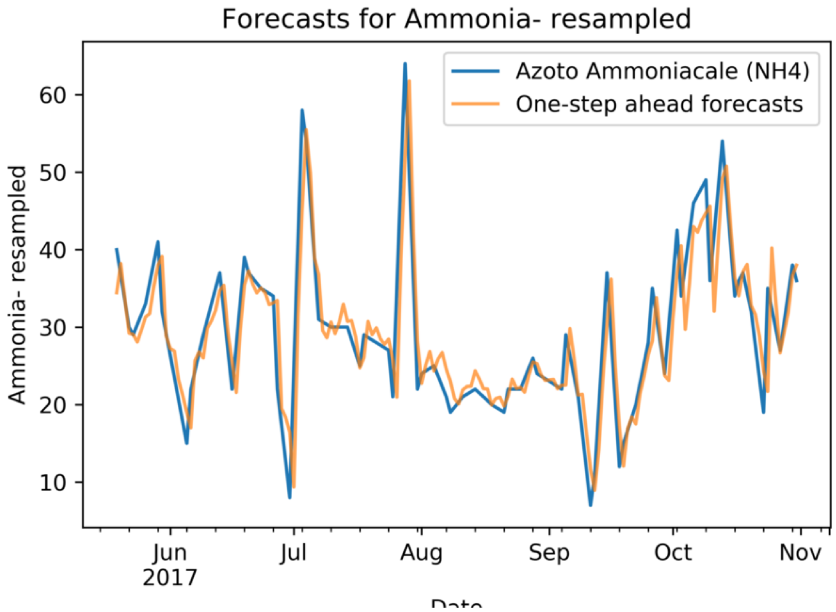


Figure 4.19: Forecasts for SARIMA model found for ammonia resampled

From table 4.41, it appears that the MSE on forecasting is considerably low. This is confirmed by figure 4.19, which shows the use of the model for prediction.

Let us analyze the diagnostics for this model, as shown in figure 4.20.

- from the histogram, it can be seen that the KDE plot resemble $N(0, 1)$. However, the fitness appears to be slightly lower than the one achieved with the original time series;
- from the normal Q-Q plot, the ordered distribution of residuals can be found to follow the linear trend of the samples taken from $N(0, 1)$, except for border values;
- no seasonal effects can be found within the standardized residuals, and the low correlation shown by the correlogram confirms that indications.

These diagnostics allow to consider the SARIMA model satisfactory enough to be used.

Remarks. A relevant result from previous analysis is that the evaluation of such models cannot rely only on a single metric, such as the AIC. As an example, in this specific case, relying only on AIC would not allow to infer that data coming from a properly-conducted acquisition campaign, with regular and (timely) dense samplings, would greatly enhance the predictive capabilities of achievable models. These considerations can be extended to all the cases which will be discussed in the following.

Nitric Oxide

Time series Exploratory Data Analysis. As reported in [95], many of the values found for nitric oxide this parameter are below the detection threshold of the instrument used for the acquisition. This is expected to be reflected on anomalies in the time series, and this is confirmed by the diagnostics shown in figure 4.21.

From this diagnostics, it is clear that:

- data hold more representative power starting from June 2017, when they have been more densely sampled. Such irregular sampling obviously undermine the descriptive capability of a model obtained on this series;
- the ACF and PACF plots do not give meaningful suggestions on underlying AR or MA processes, as there are no clear (and permanent) cutoffs for both of the functions;
- the normal Q-Q plot greatly deviates from the behavior expected for a normal distribution;
- the histogram shows a main normal distribution, which is probably related to samples below the detection threshold, and several 'tail distributions' relative to the time period where data appear to be sampled in a more meaningful way.

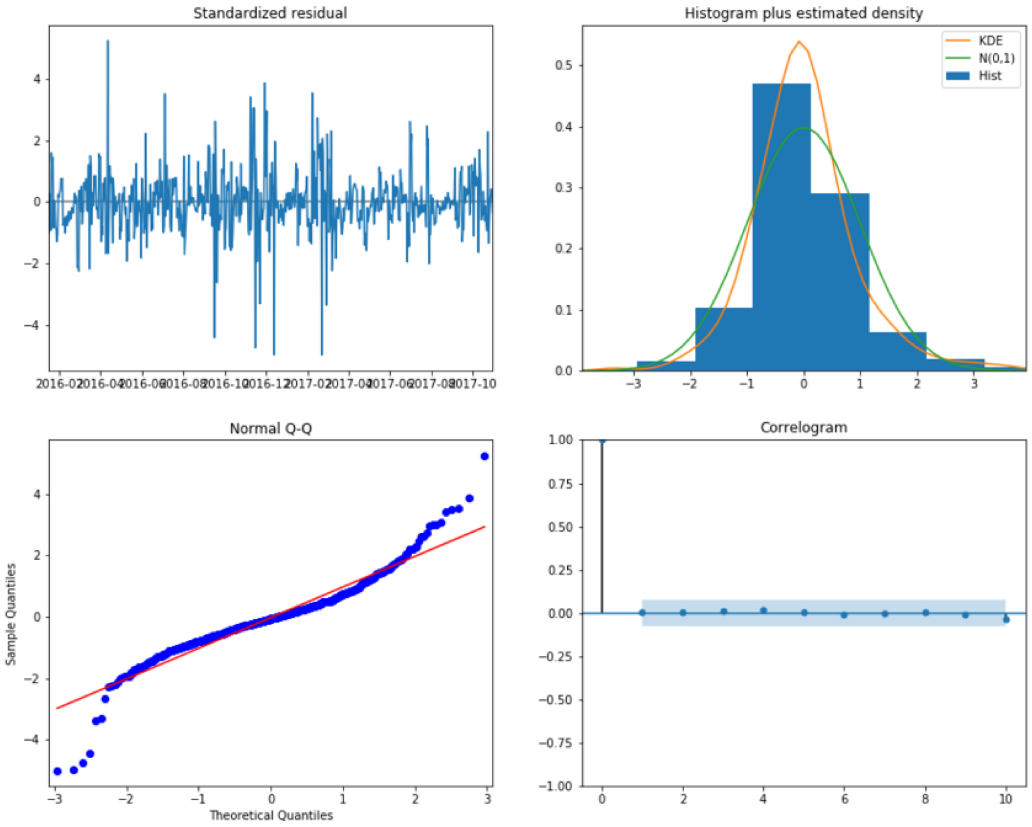


Figure 4.20: Diagnostics for SARIMA model found for ammonia resampled

The STL decomposition 4.22, is able to capture the slowly increasing trend, which is probably

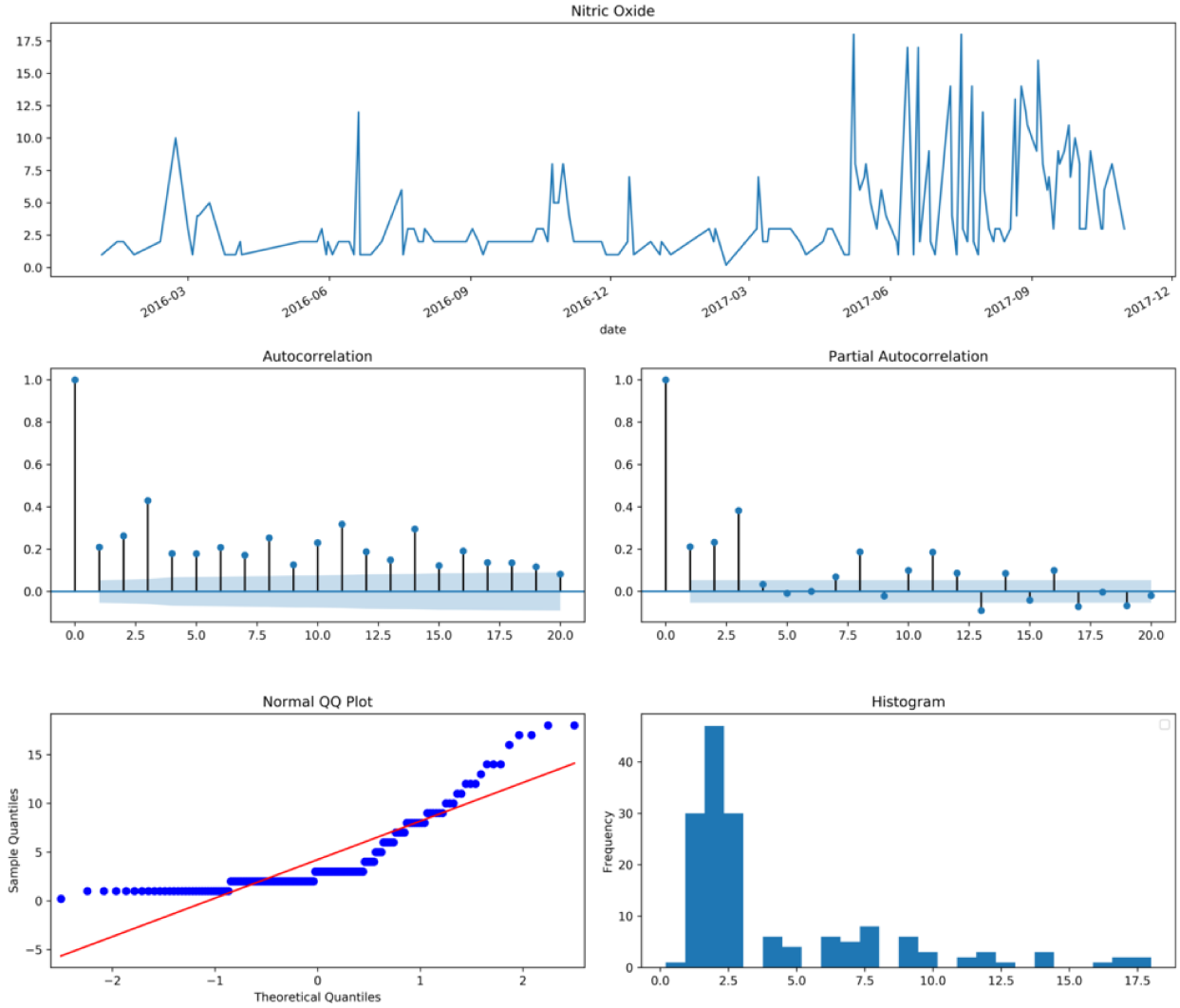


Figure 4.21: Analysis of nitric oxide for Vimercate Wastewater Treatment Plant

due either to a more dense sampling or to values which are found to be above the detection threshold of the instrument.

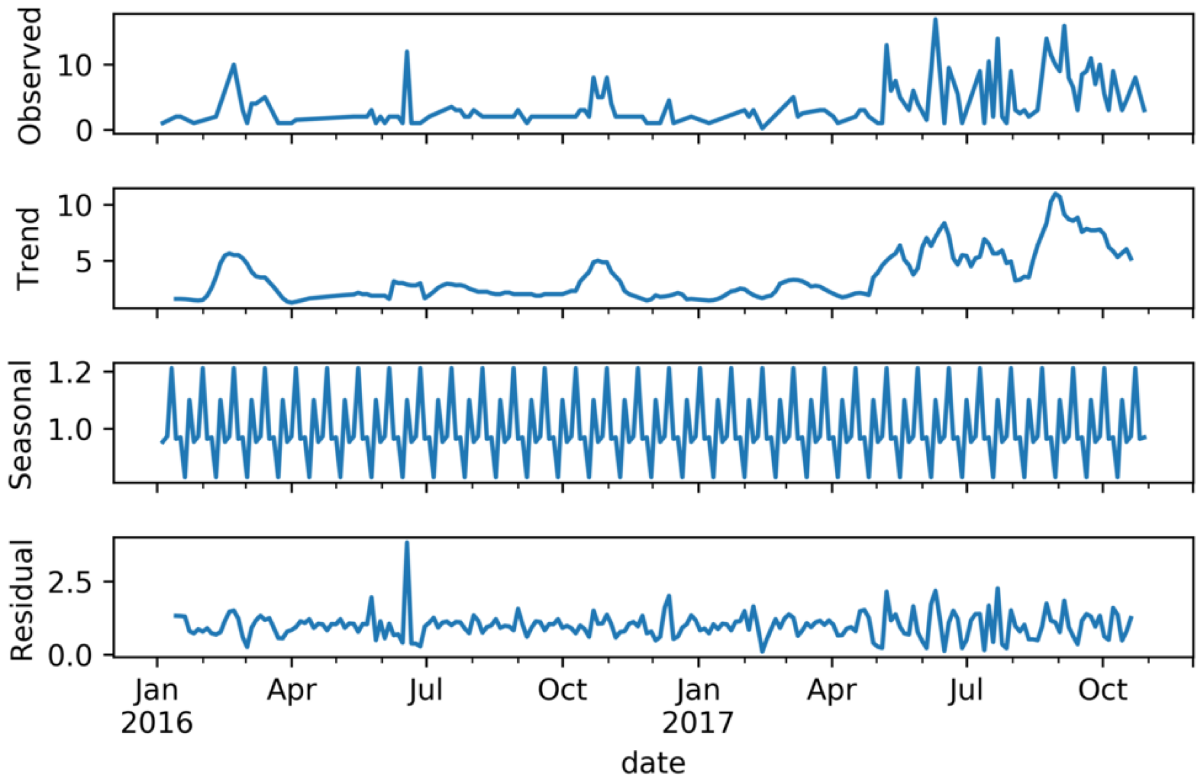


Figure 4.22: STL decomposition for nitric oxide

SARIMA modeling. Let us evaluate the best SARIMA model obtained for the original time series.

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
3	1	10	0	0	0	804.01	12.24

Table 4.42: Parameters found for best seasonal ARIMA model on nitric oxide.

Table 4.42 that, again, seasonality is not accounted by the best model. Diagnostics, shown in figure 4.23, show an overall good behavior of the model, as histogram and normal Q-Q plot adequately resemble a normal distribution (with a remarkable deviation on the borders), while correlogram shows no correlations between residuals.

However, it is clear that the model, shown in figure 4.24, cannot capture rapid spikes in data, adhering only to the overall trend of the series.

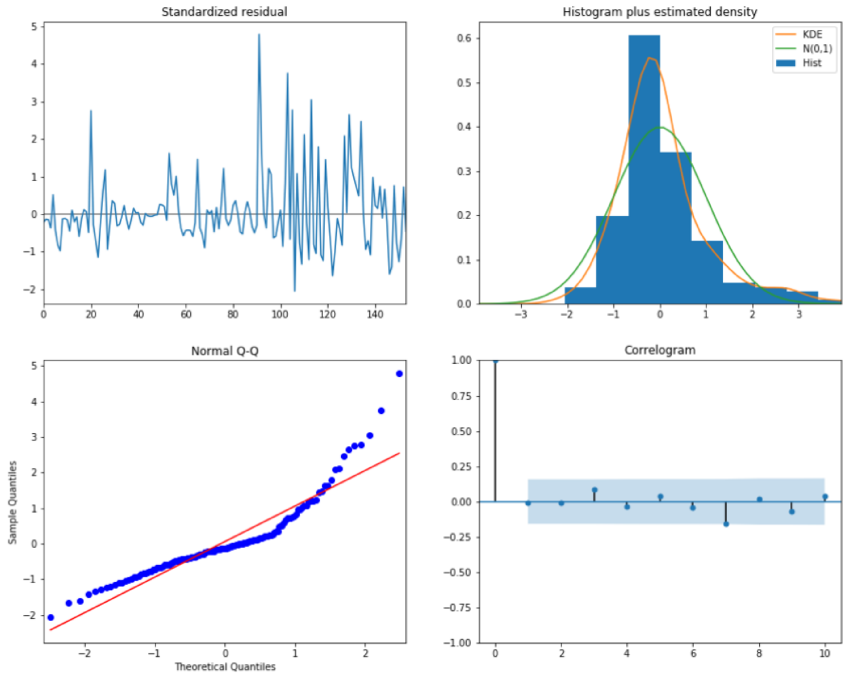


Figure 4.23: Diagnostics for SARIMA model found for nitric oxide

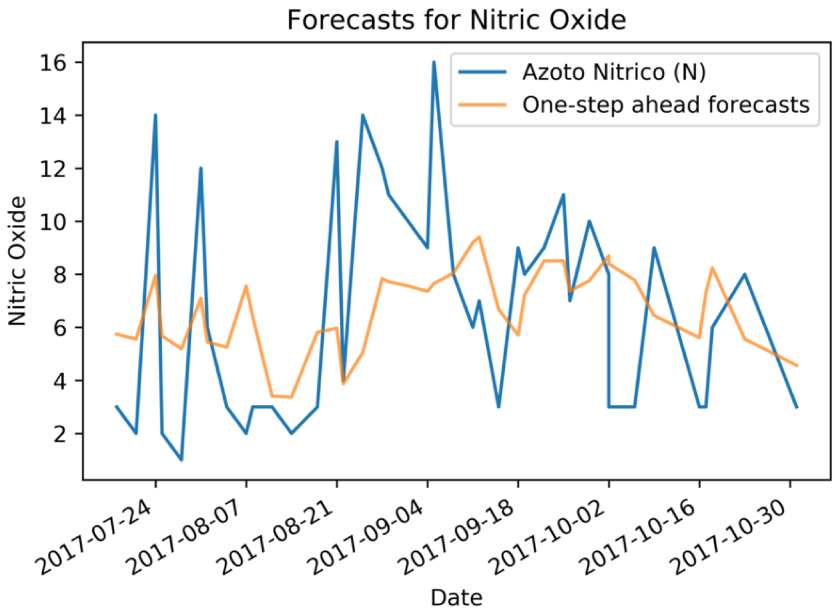


Figure 4.24: Forecasts for SARIMA model found for nitric oxide

Let us then evaluate the effect of oversampling of the time series.

SARIMA model on resampled time series. Parameters for the best SARIMA model are shown in table 4.43, and, again, indicate a higher value of AIC, if compared with the best model found on original data, with a lower mean squared error on forecasts.

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
6	1	10	0	0	0	2409.45	5.44

Table 4.43: Parameters found for best seasonal ARIMA model on resampled time series of nitric oxide.

Diagnostics, shown in figure 4.25, show an irregular behavior of the best SARIMA model: specifically, the kernel density estimation of the distribution of the residual considerably diverges from a normal distribution $N(0, 1)$, while the correlogram shows some correlations between residuals.

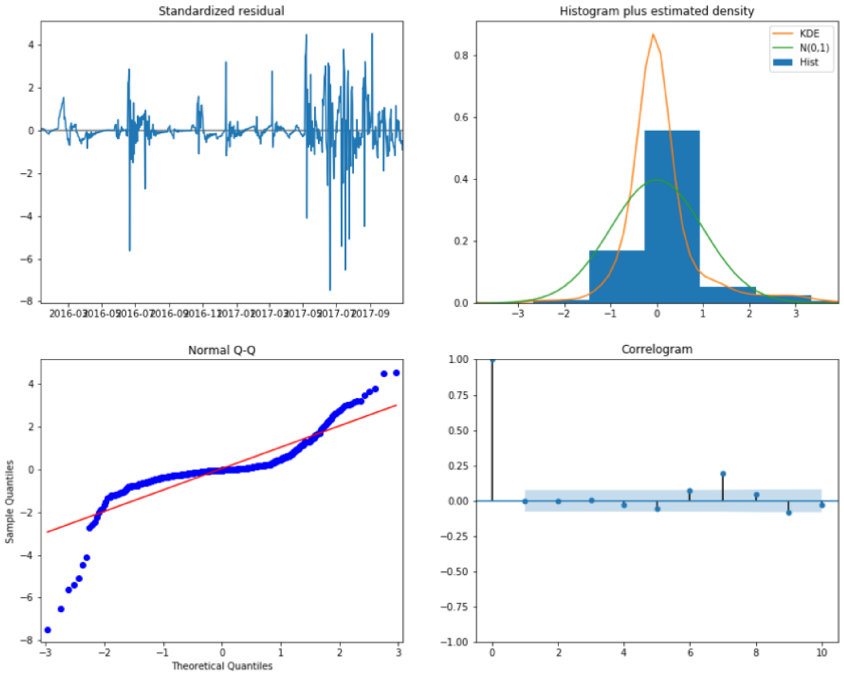


Figure 4.25: Diagnostics for SARIMA model found for nitric oxide resampled

However, the predictive capabilities of the model appears to be improved, as it can be seen from figure 4.26, and the model appears to properly characterize spikes in data.

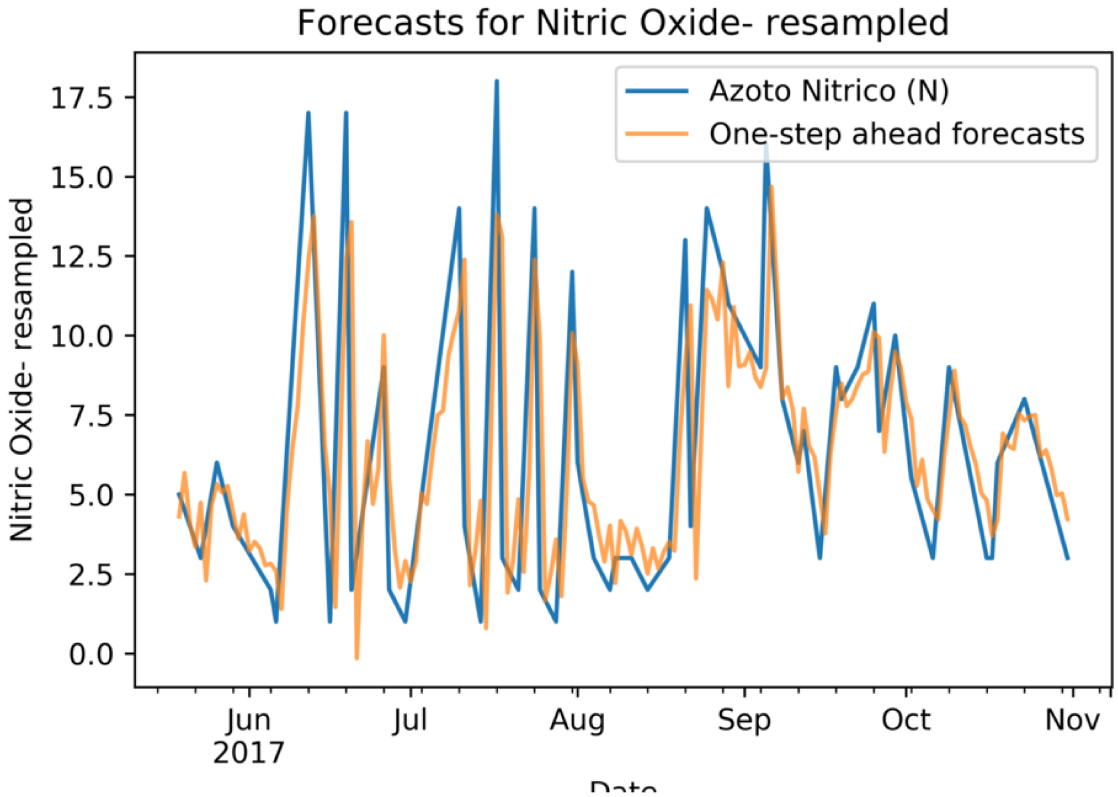


Figure 4.26: Forecasts for SARIMA model found for nitric oxide resampled

Remarks. This time series is heavily compromised by a bad designed acquisition campaign, which undermines the achievable results due to the presence of a relevant bias determined by the high (with respect to the specific scenario) detection threshold of the instrument used during the acquisition. Therefore, this should be one of the leading principles of future acquisition campaigns.

Nitrous Oxide

Time series Exploratory Data Analysis. Data for nitrous oxide appear to be more densely sampled with respect to data for nitric oxide; however, this series also presents some anomalies, as shown by the diagnostics depicted in figure 4.27. Specifically, the ACF and PACF suggests an ARMA behavior, due to the gradual tailing off of both the functions; however, there are some spikes at samples with high lag, which somehow weaken this suggestion. Furthermore, the normal Q-Q shows considerable deviations from normality at the borders, and effects related

to an inadequate detection threshold can be found at the lower border of the plot. This is confirmed by the histogram, which clearly suggests that data distribution is not normal, and there is a bias effect related to a low detection threshold.

As for STL decomposition, shown in figure 4.28, there are no clear indications of trends within data.

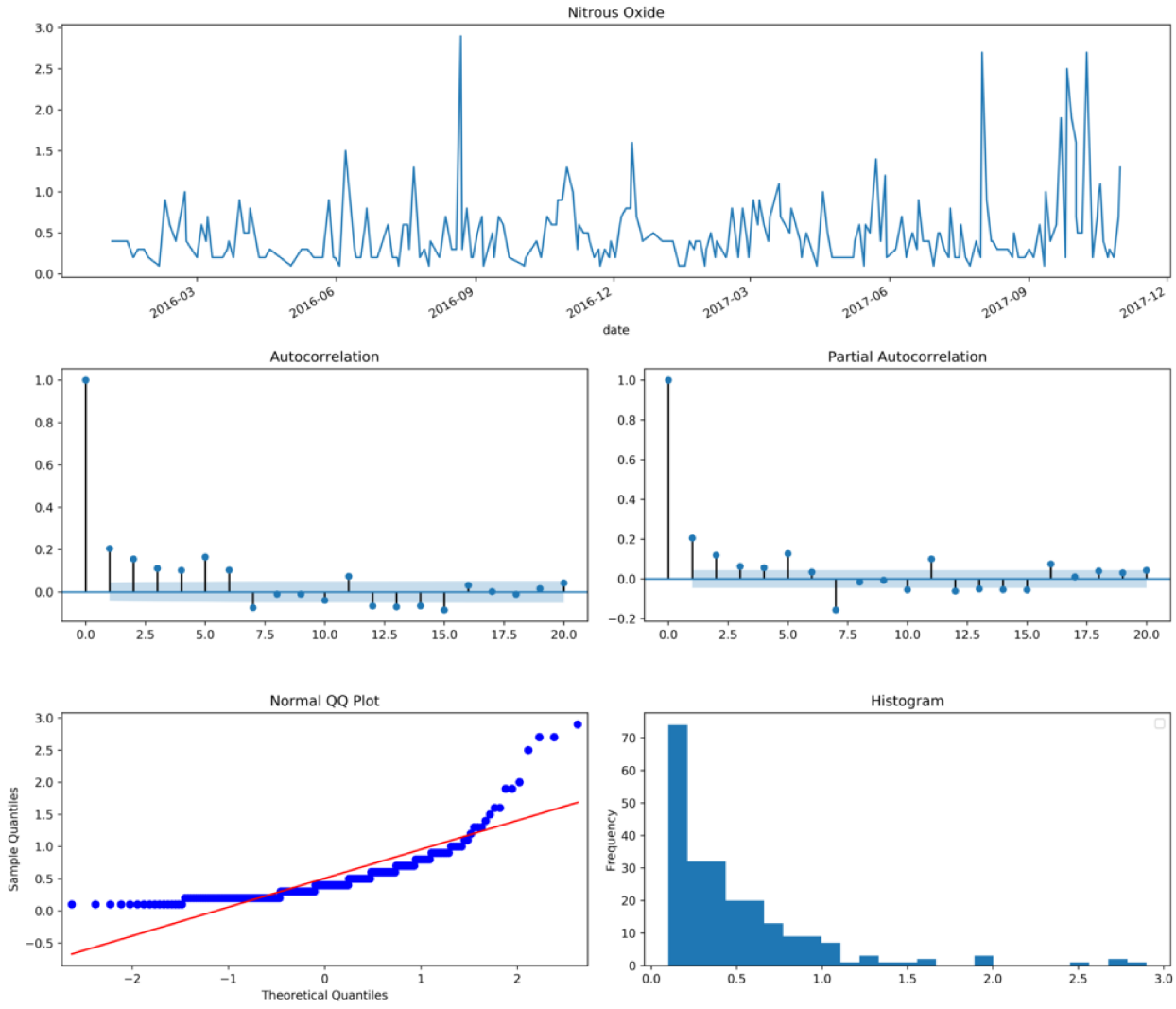


Figure 4.27: Analysis of nitrous oxide for Vimercate Wastewater Treatment Plant

SARIMA modeling. Parameters of the best SARIMA model found on nitrous oxide data are shown in table 4.44.

As expected, seasonal effects are not taken into account by the model. Diagnostics, shown in figure 4.29, highlights that residuals resemble a normal distribution, and the normal Q-Q plot

shows the usual behavior, following a normal distribution except for considerable deviations on border values. As for the correlogram, it appears that no relevant correlations exist between residuals.

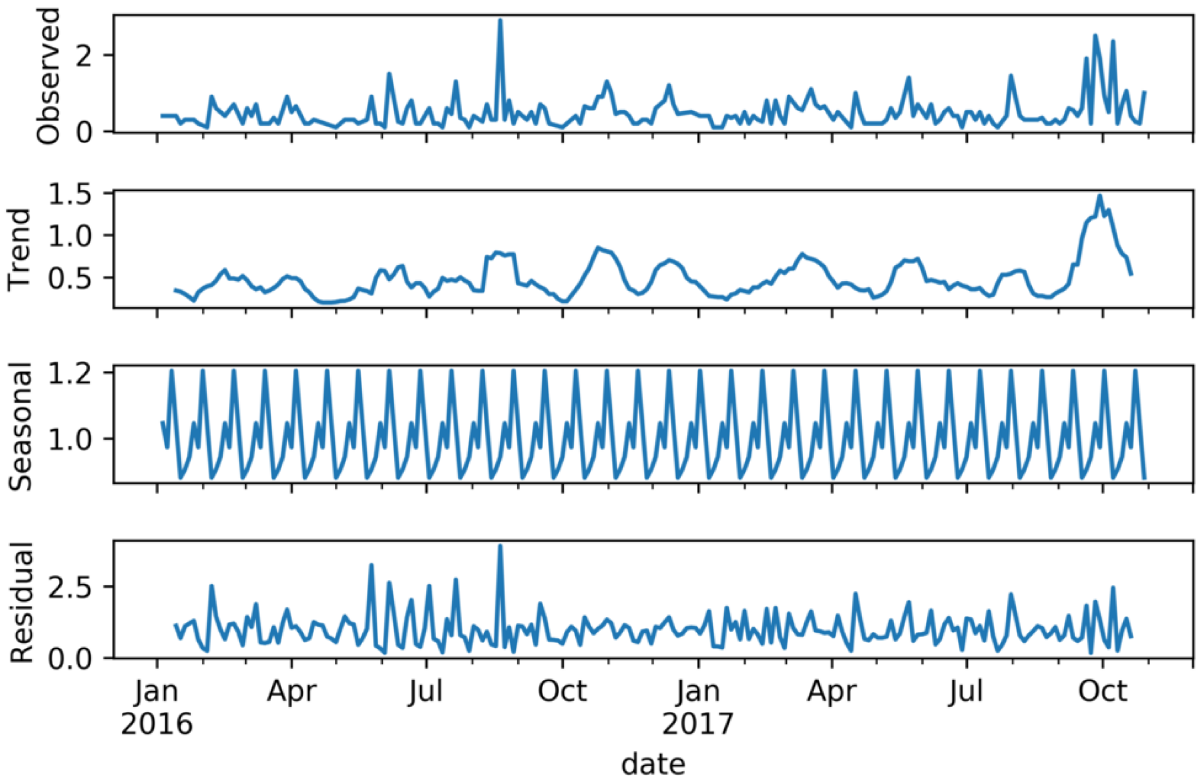


Figure 4.28: STL decomposition for nitrous oxide

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
1	1	1	0	0	0	281.69	0.39

Table 4.44: Parameters found for best SARIMA model on data acquired for nitrous oxide.

The MSE on forecasts, from table 4.44, appears to be low; however, this is due to the values assumed by the parameter itself and, as it is clear from figure 4.32, the model which has been found is not able to follow the rapid variations of the time series.

SARIMA modeling on resampled series. Results for the best SARIMA model achievable on the resampled version of the time series are shown in table 4.45.

Interestingly, this is the first situation where a negative value (which is admissible) for the AIC is found. Furthermore, the best SARIMA model for this series does not envisage for a

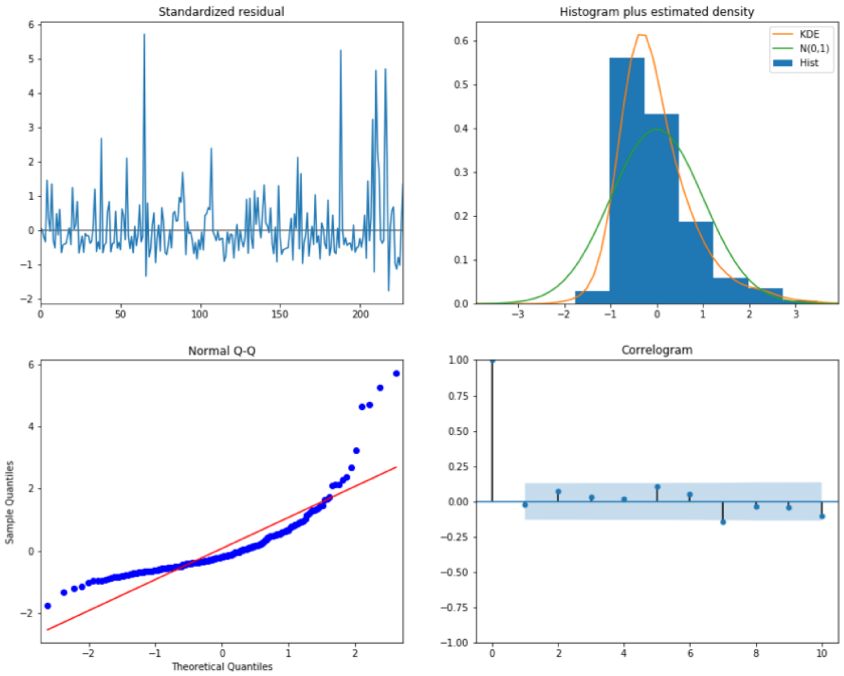


Figure 4.29: Diagnostics for SARIMA model found for nitrous oxide

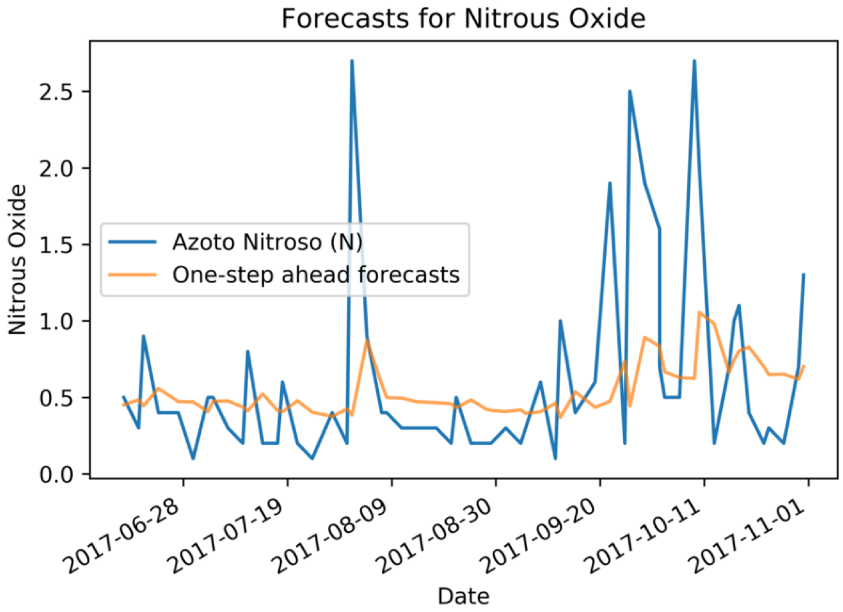


Figure 4.30: Forecasts for SARIMA model found for nitrous oxide

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
3	0	3	0	0	0	-61.43	0.12

Table 4.45: Parameters found for best SARIMA model on resampled nitrous oxide.

trend component within data. Diagnostics are shown in figure 4.31, and residuals appear to adequately fit a normal distribution, with no correlation effects.

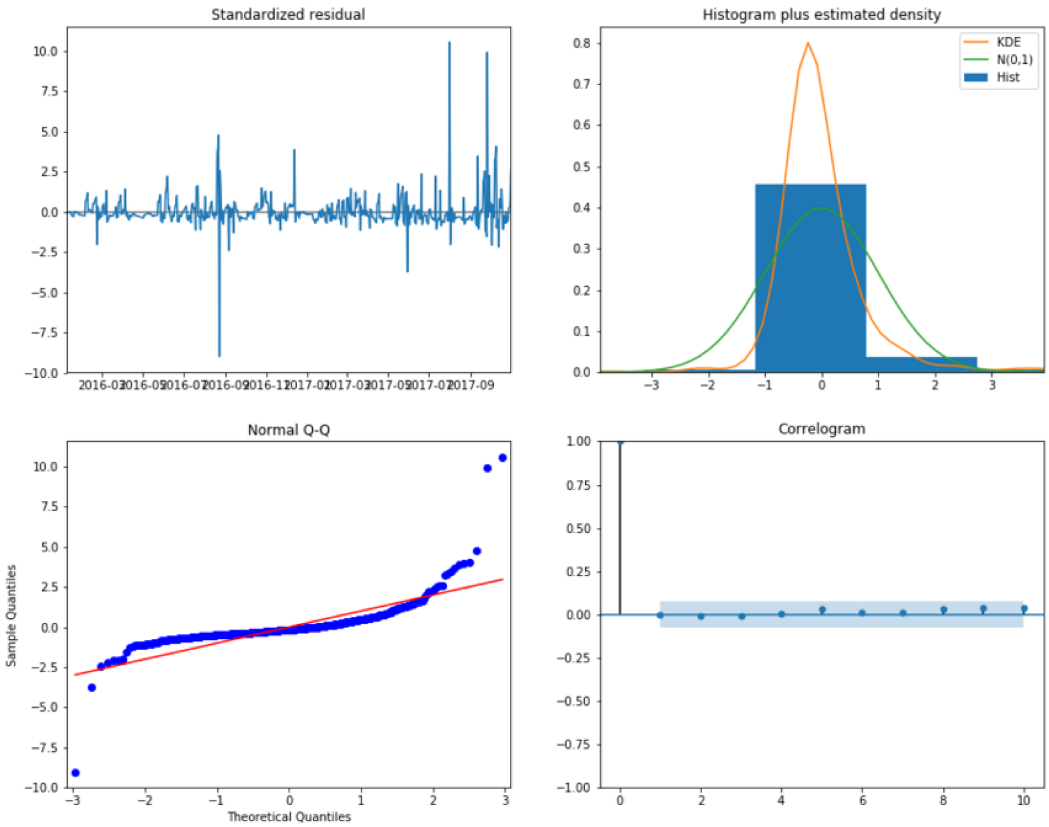


Figure 4.31: Diagnostics for SARIMA model found for nitrous oxide resampled

Forecasts shown in figure 4.32 show the suitability of the model to make forecasts on the resampled time series.

Remarks. Interestingly, the differences in the MSE for forecasts between the original and the resampled time series are not relevant. Therefore, one, by just looking at the numeric value, would assume that no relevant improvements can be achieved by oversampling the time series. However, forecast plots tell a different story: the model found on the original time series cannot model any of the complexity of the underlying process, while the model found on the resampled series can. This is a perfect showcase of the power of exploratory data analysis.

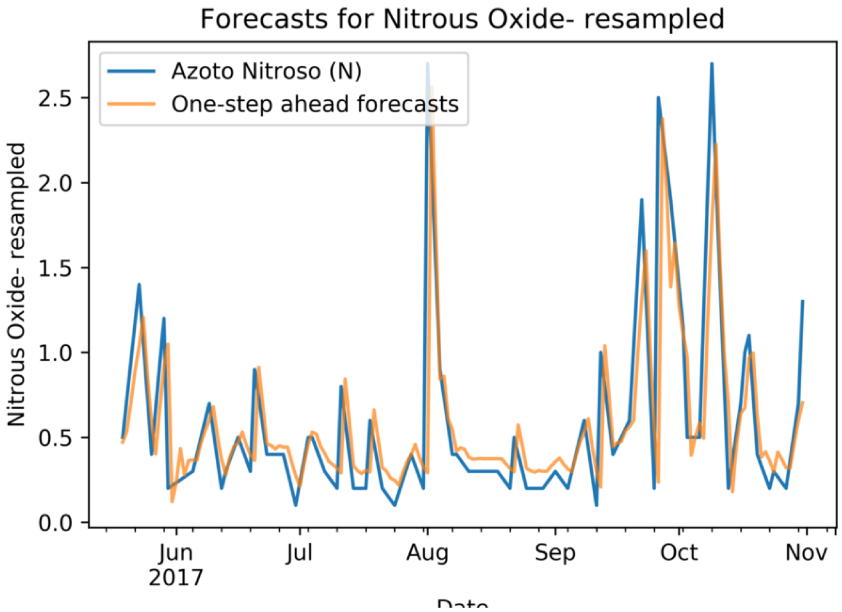


Figure 4.32: Forecasts for SARIMA model found for nitrous oxide resampled

4.6.2 Chemical Oxygen Demand

Chemical considerations. COD shows high variability, and is often found to be above the legal threshold value of 500 mg/l, with several peaks above the 1000 mg/l. Chemical analysis specify that this is the only parameter considered relevant to the real time evaluation of the presence of oxygen, as the time needed to determine the BOD is not compatible with real-time requirements [95].

Time series Exploratory Data Analysis. The time series relative to COD is shown in figure 4.33. In this case, both the ACF and PACF functions show a gradually decreasing tail, which suggests that the underlying process is an ARMA. Furthermore, the normal Q-Q plot and the histogram resemble a normal distribution, even if it is possible to infer from the histogram a high skewness, since the distribution does not appear to be symmetrical around the mean value.

It is also not possible to infer a clear trend from the STL decomposition, as shown in figure 4.34.

SARIMA modeling. Parameters for the best SARIMA model found for COD data are reported in table 4.46.

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
0	1	4	0	0	0	3569.5	40566

Table 4.46: Parameters found for best SARIMA model on COD.

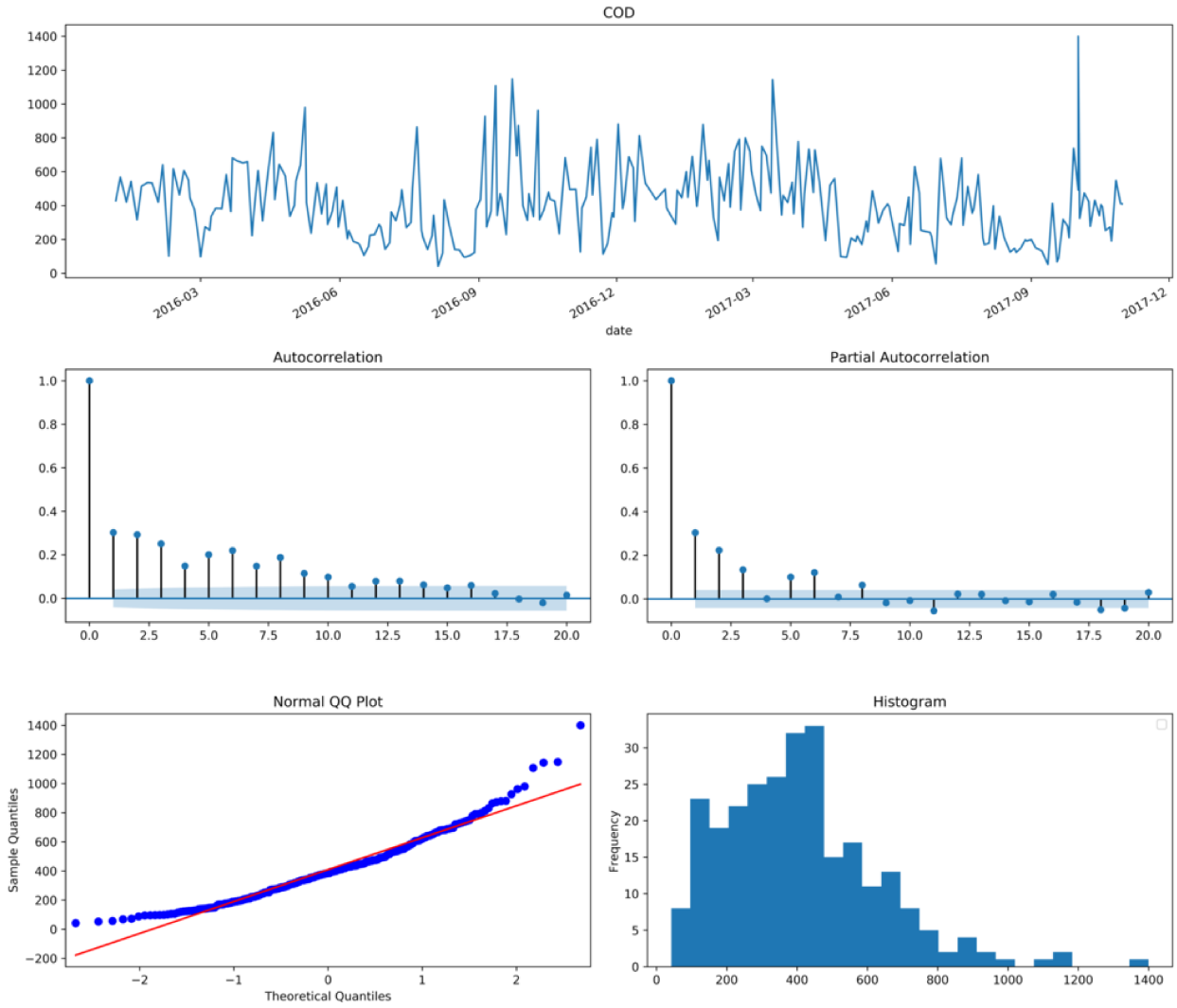


Figure 4.33: Analysis of COD for Vimercate Wastewater Treatment Plant

Interestingly, no suggestions of an AR process are found within the model. This can be addressed while extending the range of values used for the grid search (which have been limited to a maximum value of 10); however, such extension must also envisage for an increased computational load for grid search, which could not be addressed with the currently available hardware equipment.

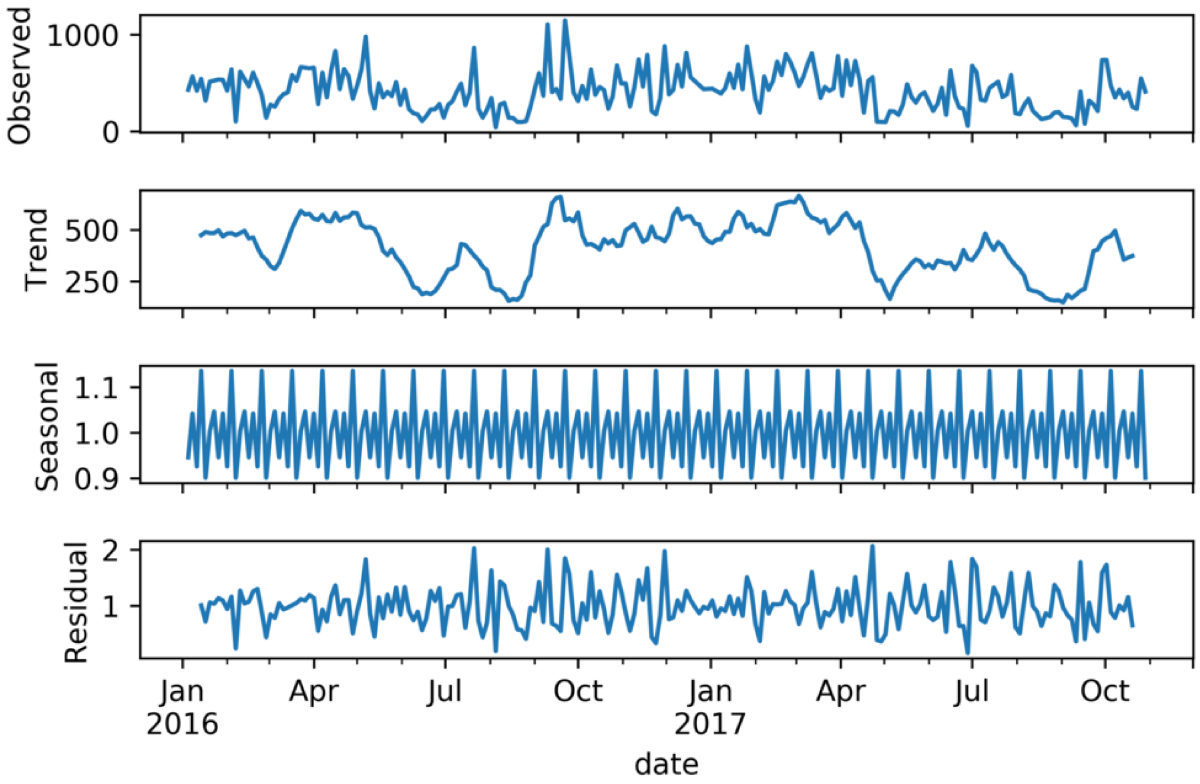


Figure 4.34: STL decomposition for COD

Diagnostics, shown in figure 4.35, show how residuals are a good fit for a normal distribution $N(0, 1)$, and that there are no correlations between them. However, forecasts show that, again, the number of samples given to the SARIMA model are not adequate for the modeling of sudden spikes in the time series.

SARIMA modeling on resampled time series. Parameters for the best SARIMA model found for the resampled version of COD are reported in table 4.47.

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
4	1	4	0	0	0	8033.69	7579.27

Table 4.47: Parameters found for best SARIMA model on COD resampled.

Interestingly, the order of the AR component suggests that oversampling allows to highlight this part of the process, which, again, was expected from the ACF and PACF plots shown in the evaluation of the time series.

Diagnostics shown in figure 4.37 highlight a proper behavior of the residuals of the time series,

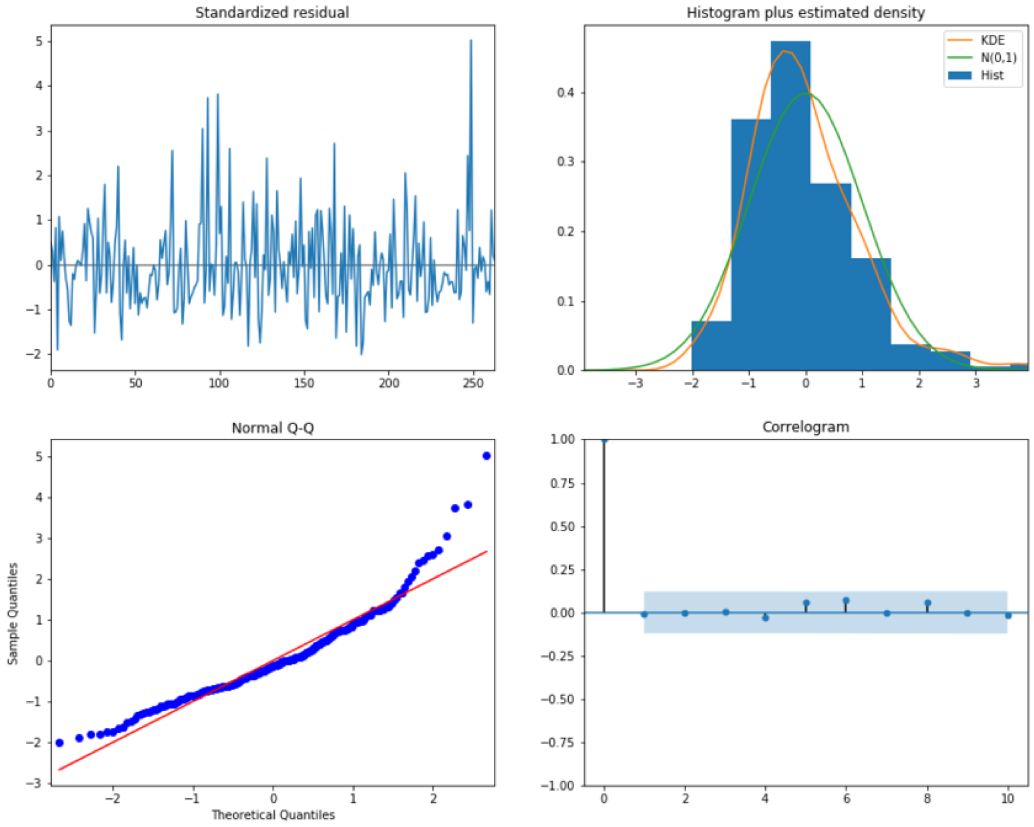


Figure 4.35: Diagnostics for SARIMA model found for COD

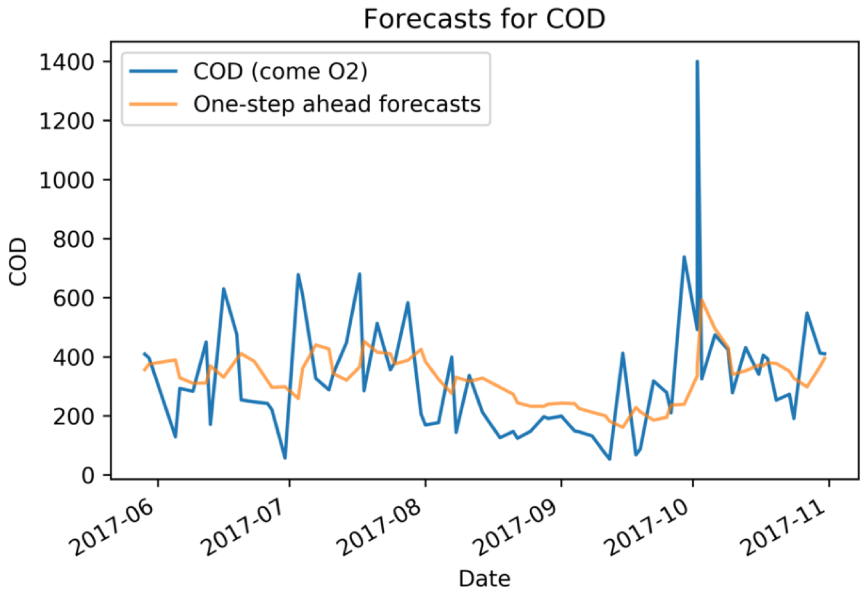


Figure 4.36: Forecasts for SARIMA model found for COD

even though high lags in the correlogram may suggest some form of correlation between residuals. However, as expected, the model is now able to follow quick variations in the values of COD (figure 4.38).

Remarks. Also this case shows the importance of exploratory data analysis: relying only on the numeric value of MSE, one may assume that the capabilities of the SARIMA model are limited. However, plots quickly highlight that the values obtained for the MSE also depends on the values assumed by COD itself.

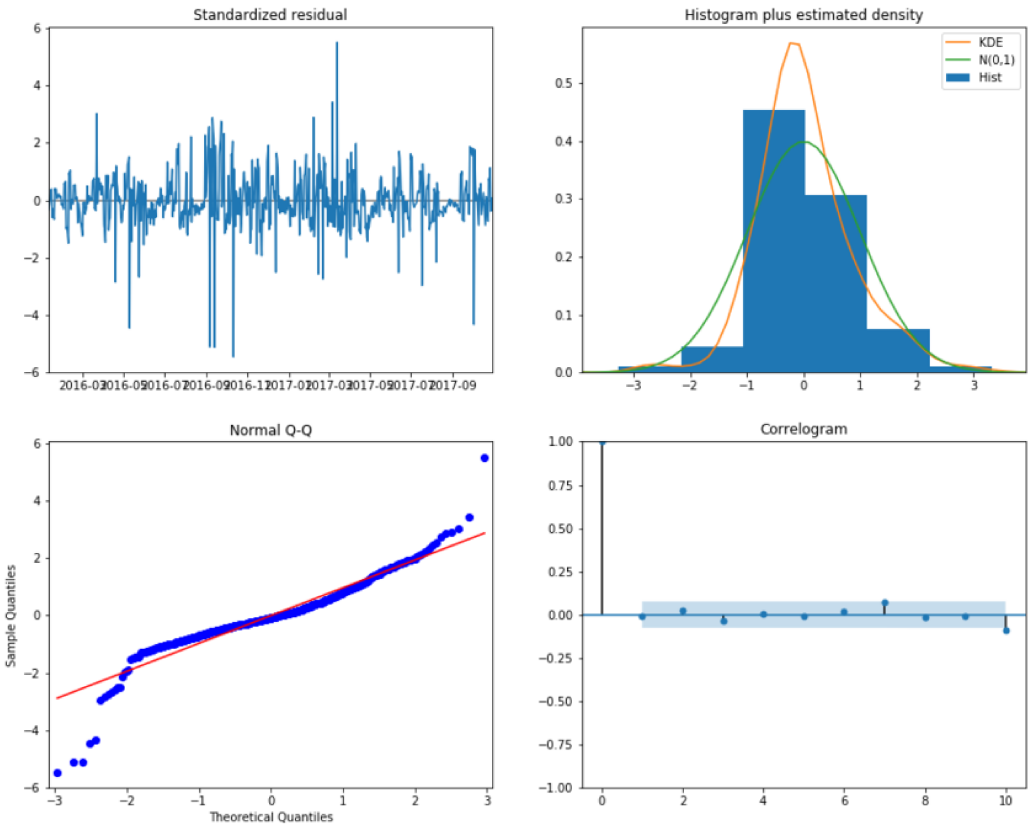


Figure 4.37: Diagnostics for SARIMA model found for COD resampled

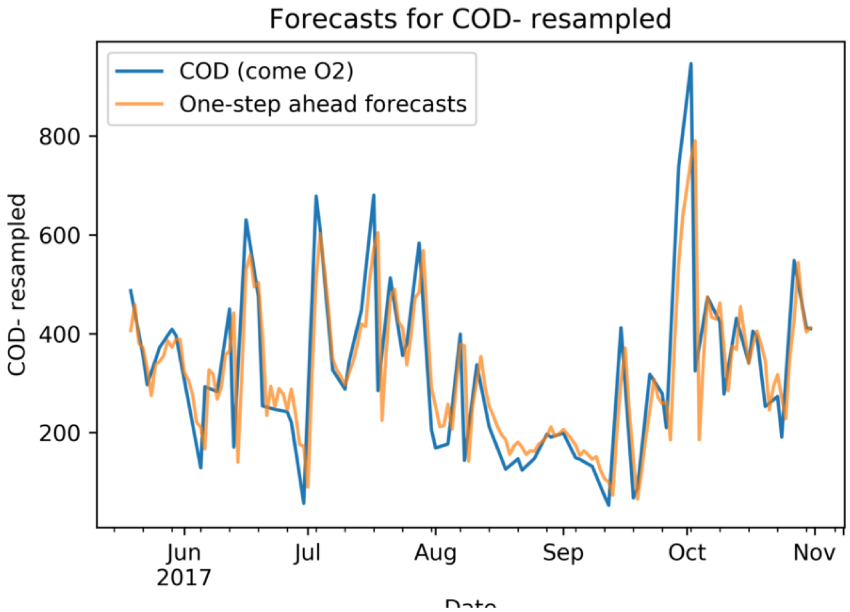


Figure 4.38: Forecasts for SARIMA model found for COD resampled

4.6.3 Chloride

Chemical considerations. Chloride varies around an average value of 175 mg/l, with some peaks during the first part of 2016; however, these values are always been found to be below the legal threshold. Authors in [95] suggest, as a future extension, to evaluate the concentration of active chloride, due to its impact on activated sludges.

Time series Exploratory Data Analysis. Chloride also shows the characteristics of an ARMA process, at it can be seen from the ACF and the PACF in figure 4.39. However, the decay of tails of both functions is combined with several consequent peaks. In this case, the normal Q-Q plot and the histogram confirm that data are normally distributed.

As for the STL decomposition, shown in figure 4.40, it does not define a clear trend.

SARIMA modeling. Parameters for the best SARIMA model found for chloride data are reported in table 4.48.

The best SARIMA model highlights the presence of the underlying AR and MA processes. Residuals optimally fit a normal distribution, and no correlations are found by the correlogram.

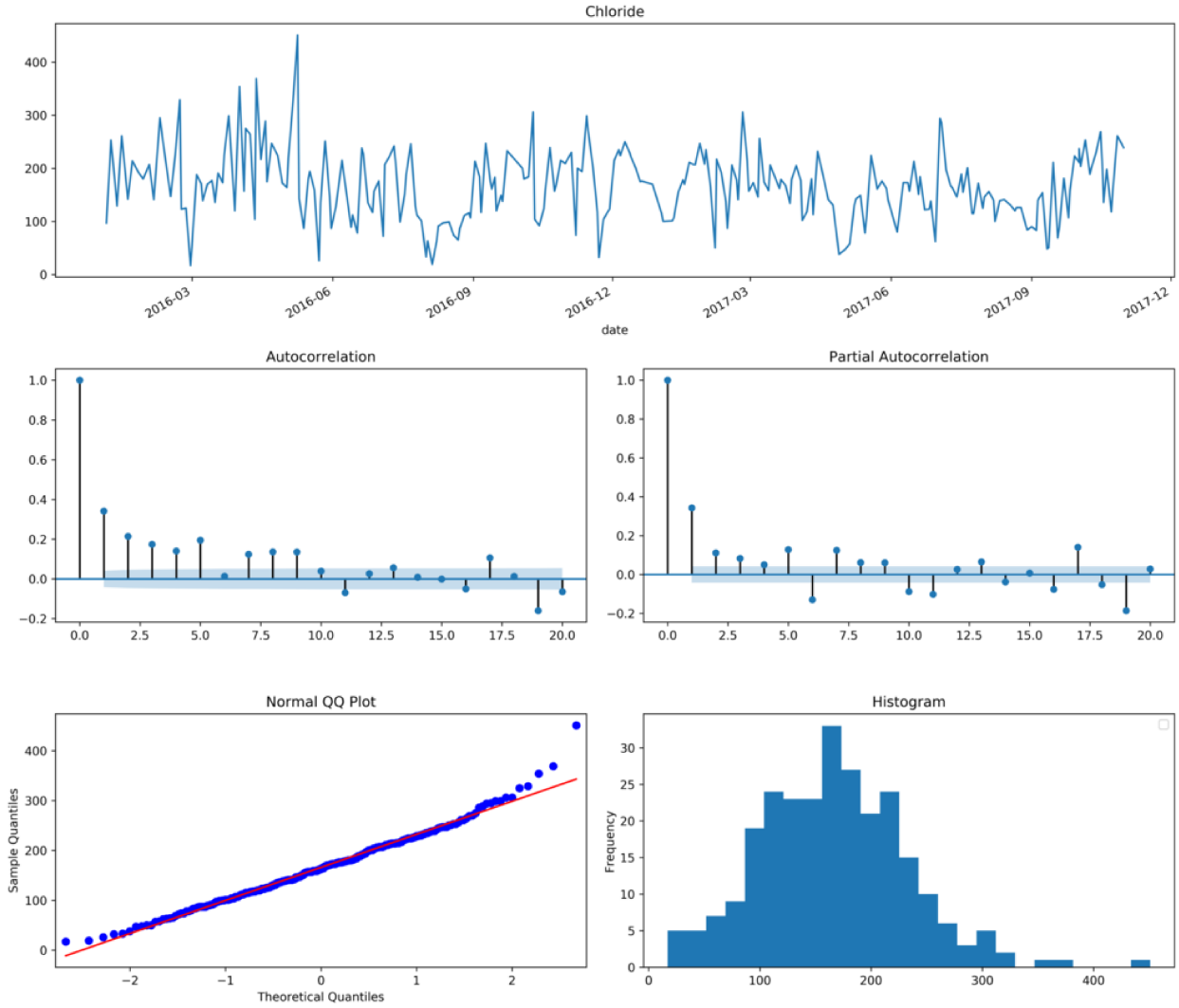


Figure 4.39: Analysis of the chloride samples over time for Vimercate Wastewater Treatment Plant

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
2	1	4	0	0	0	2866.77	2663.55

Table 4.48: Parameters found for best SARIMA model on COD.

As expected, also in this case, the model is able to follow the overall data trend, but not to capture the sudden variations shown by the process.

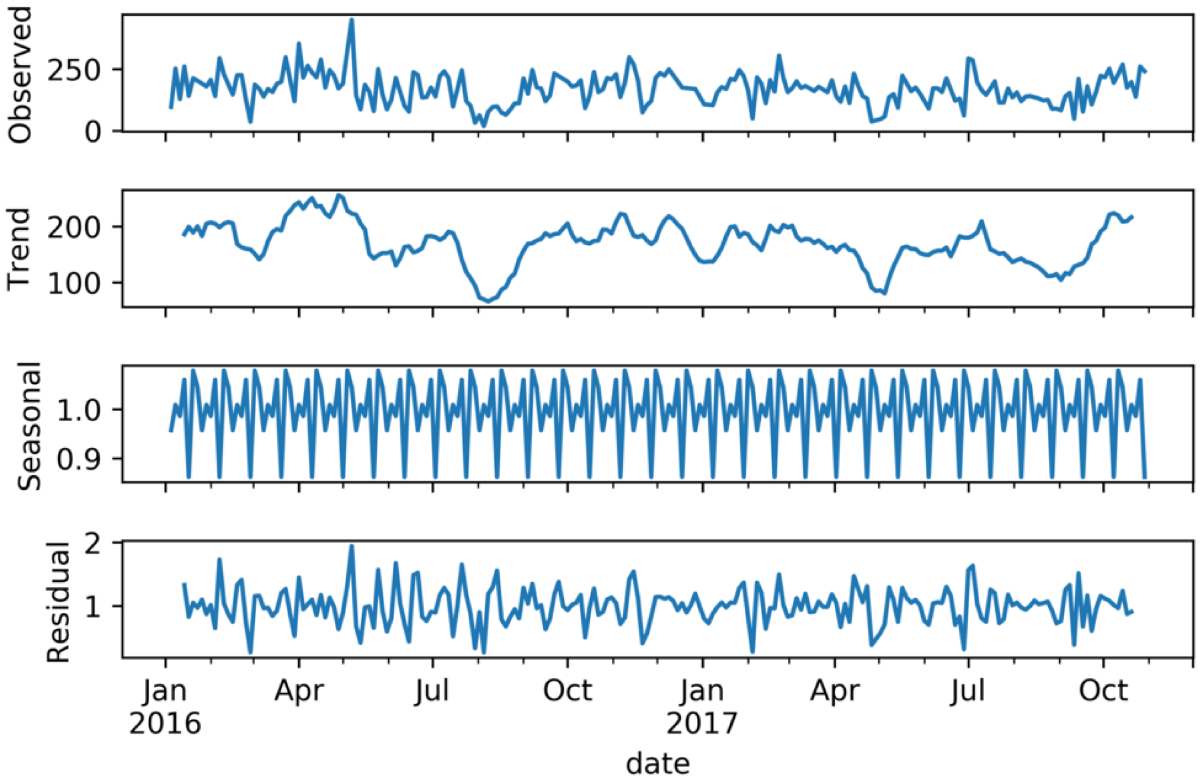


Figure 4.40: STL decomposition for chloride

SARIMA modeling on resampled time series. Parameters for the best SARIMA model found for the resampled version of chloride are reported in table 4.49.

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
3	1	4	0	0	0	6454.11	442.18

Table 4.49: Parameters found for best SARIMA model on chloride resampled.

Diagnostics for this model are shown in figure 4.43, and, also in this case, residuals show normal behavior and no relevant correlation over time.

Remarks. Interestingly, the orders found for the best SARIMA model on the original time series, and the ones for the corresponding model on the oversampled time series, are almost the same, with a slight difference in the order of the AR component. This may apparently suggest

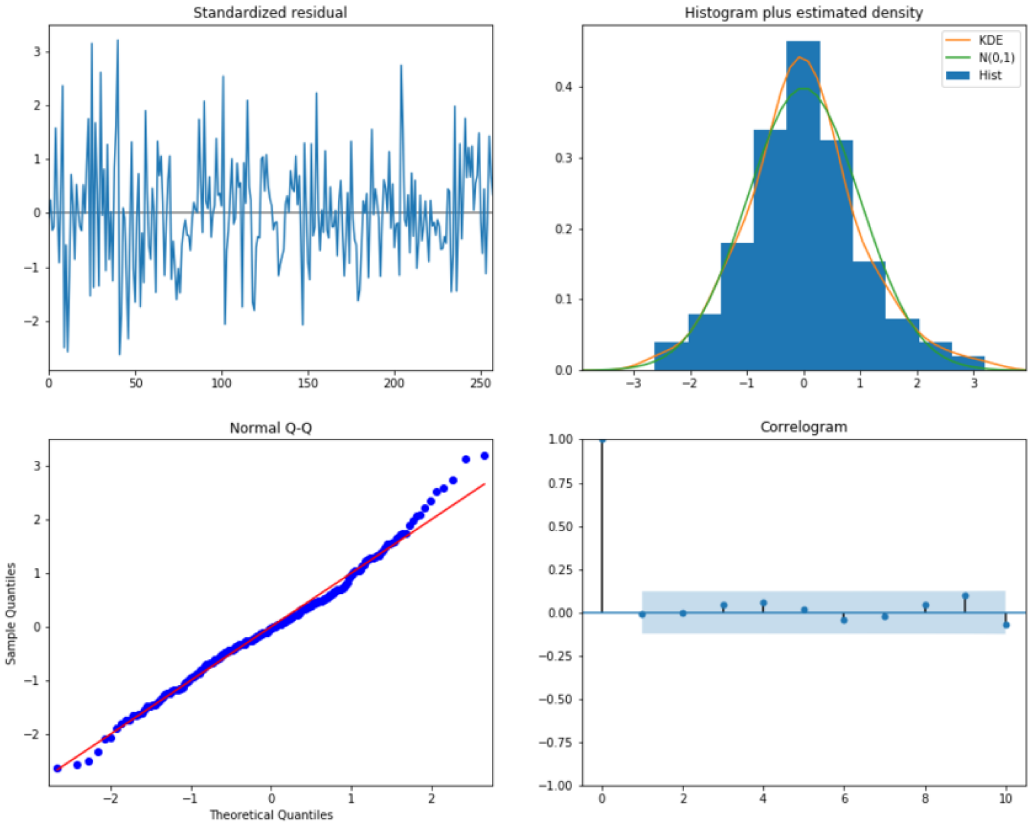


Figure 4.41: Diagnostics for SARIMA model found for chloride

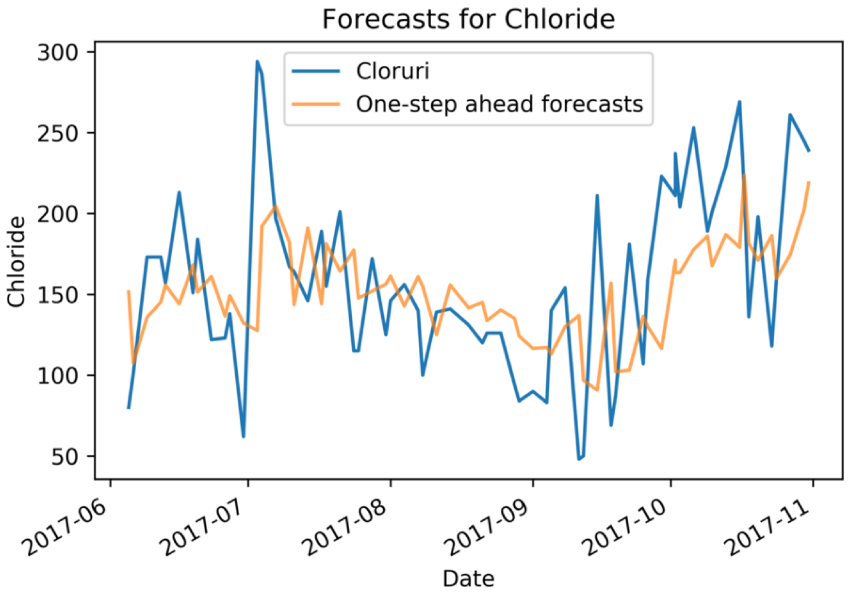


Figure 4.42: Forecasts for SARIMA model found for chloride

that, if a SARIMA with $p = 3$ is used on the original time series, improved results can be achieved. However, this is not true: first, grid search always return the best model, according to the AIC, for a set of data and, as a consequence, a SARIMA with $p = 2$ outperforms a SARIMA with $p = 3$ on the original time series. Also, effects of the interpolation must be considered: even if the original and the oversampled processes are related, they are not numerically equivalent, as the oversampled more values that, in the original, are missing.

As expected, oversampling significantly improves prediction results also in this case, as shown in figure 4.44.

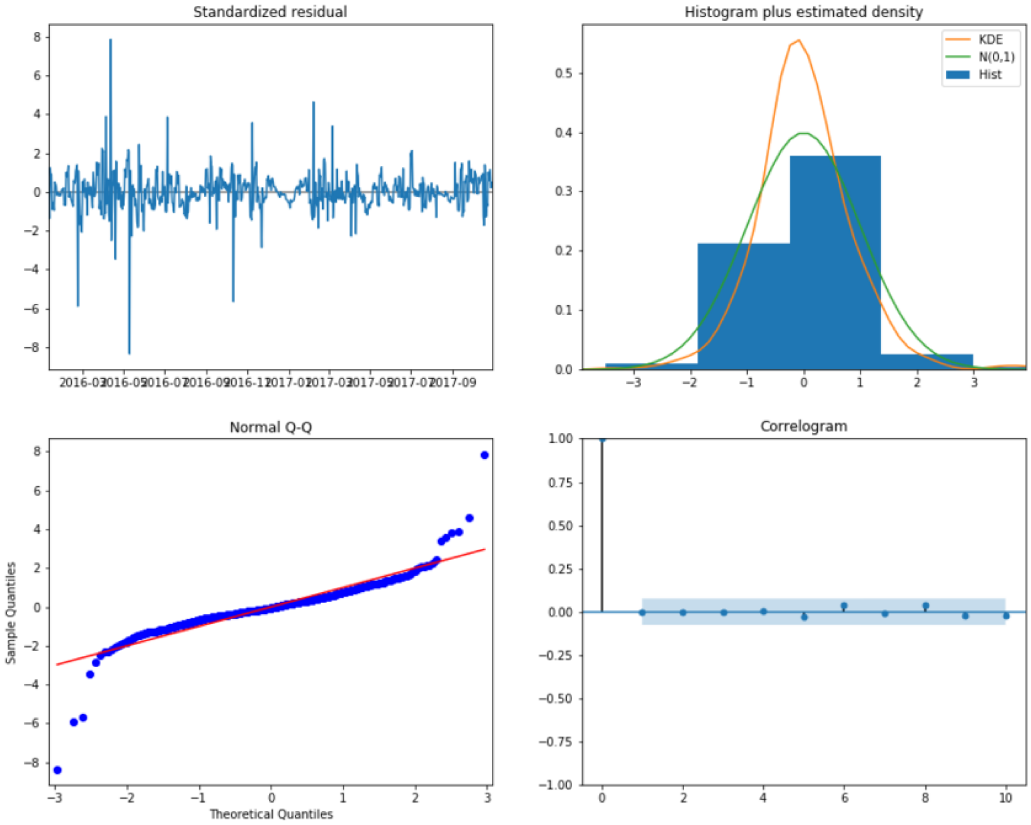


Figure 4.43: Diagnostics for SARIMA model found for chloride resampled

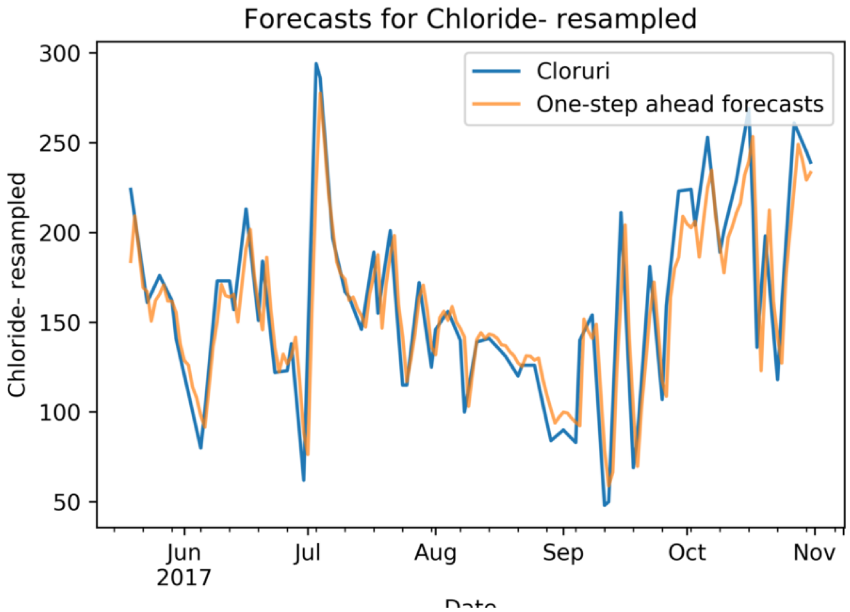


Figure 4.44: Forecasts for SARIMA model found for chloride resampled

4.6.4 Phosphor

Chemical considerations. Phosphor fluctuates around the legal limit of 10 mg/l. This value is referred to the *total* phosphor, which is made by an *organic* part, which is due to natural, biological processes, and an *inorganic* part, related to chemical-physical processes, possibly derived by anthropogenic sources. In [95], analysis underline that the total phosphor is not an indicator of the status of the wastewater; hence, suggestions are to take, in the future, further efforts to discriminate between the organic and inorganic components.

Time series Exploratory Data Analysis. Phosphor shows a behavior which resembles the one assumed by chloride. Specifically, the ACF and PACF plots suggest As for chloride, phosphor shows a more regular behavior, as it can be seen from the normal Q-Q plot and the histogram in figure 4.45. However, it must be noted that also the distribution of this compound appears to be skewed. As for the behavior of the time series, an AR process is expected, as the ACF and PACF functions resemble such behavior, as shown in chapter 2.

Also in this case, the STL decomposition shown in figure 4.46 does not highlight any global trend.

SARIMA modeling. Parameters for the best SARIMA model found for phosphor data are reported in table 4.50.

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
0	1	4	0	0	0	1487.73	10.35

Table 4.50: Parameters found for best SARIMA model on phosphor.

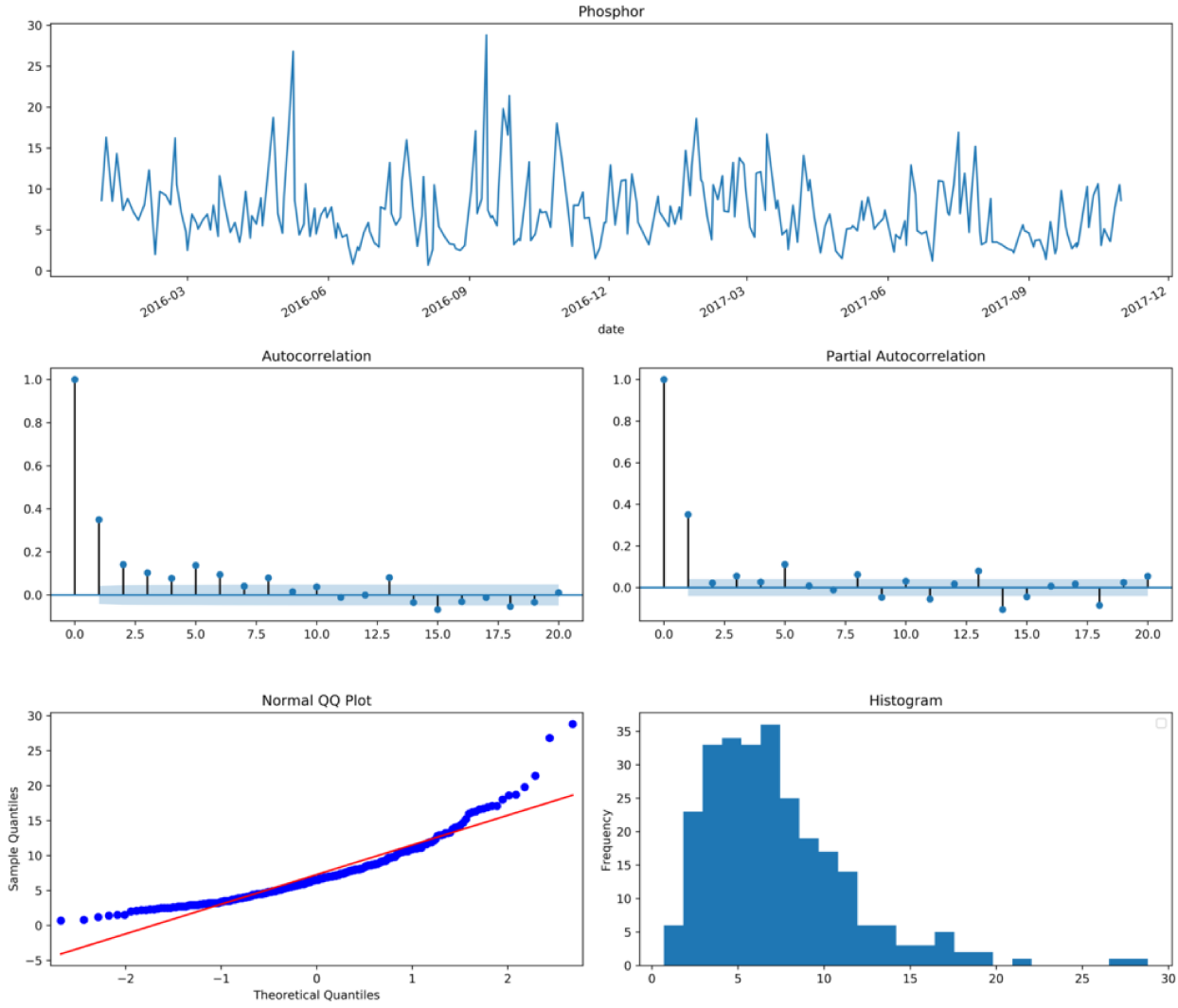


Figure 4.45: Analysis of phosphor for Vimercate Wastewater Treatment Plant

In this case, diagnostics shows that residuals are slightly skewed (figure 4.47). However, no correlations are found between them.

Again, the found SARIMA model lacks the capability to model sudden variations in the value of original data.

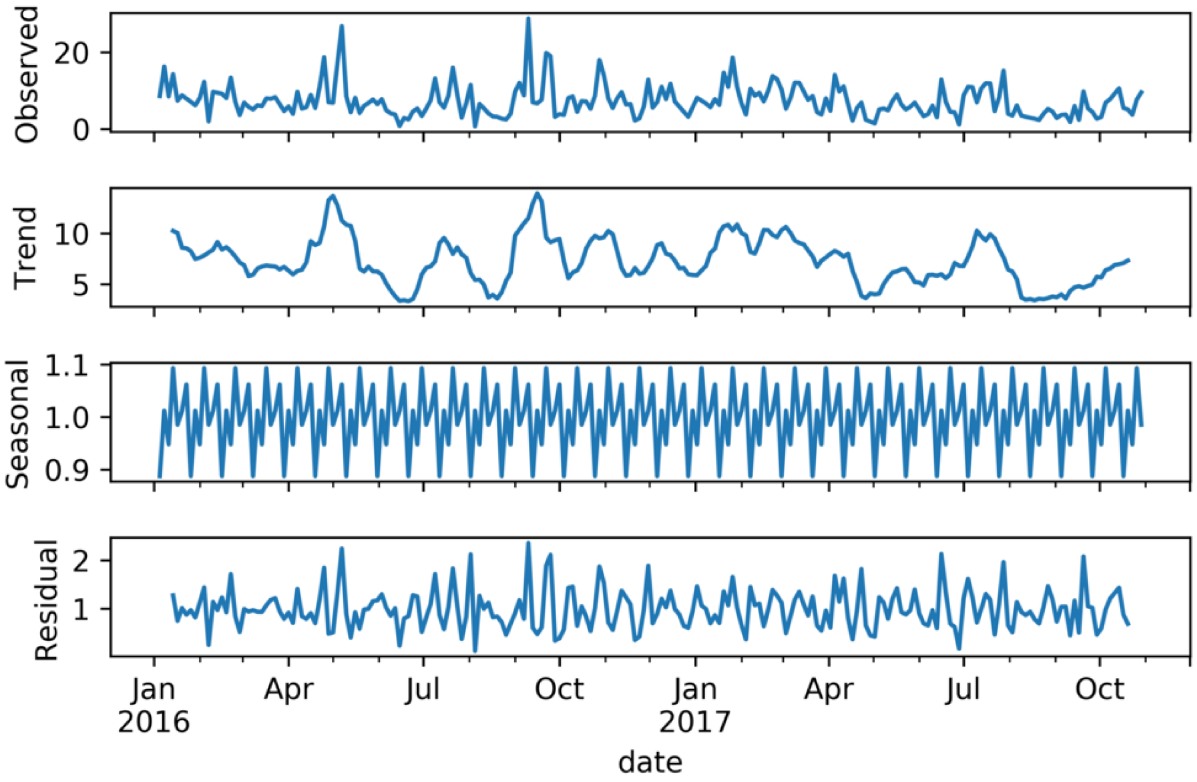


Figure 4.46: STL decomposition for phosphor

SARIMA modeling on resampled time series. Parameters for the best SARIMA model found for the resampled version of phosphor are reported in table 4.47.

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
1	1	4	0	0	0	2867.23	2.51

Table 4.51: Parameters found for best SARIMA model on phosphor resampled.

Diagnostics resemble results achieved by the SARIMA model for the original time series, even if the skewness appear to be less emphasized.

Forecasts, as expected, show a proper behavior on the resampled time series.

Remarks. No particular remarks, apart from the one previously found, can be made on this time series.

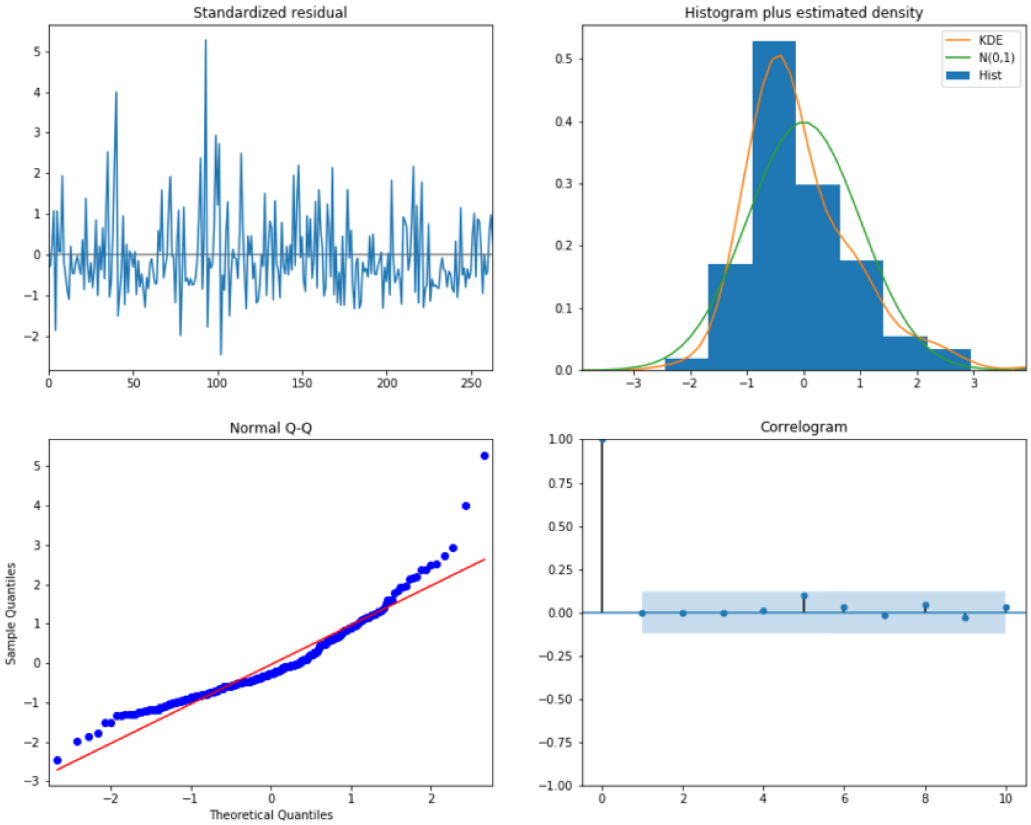


Figure 4.47: Diagnostics for SARIMA model found for phosphor

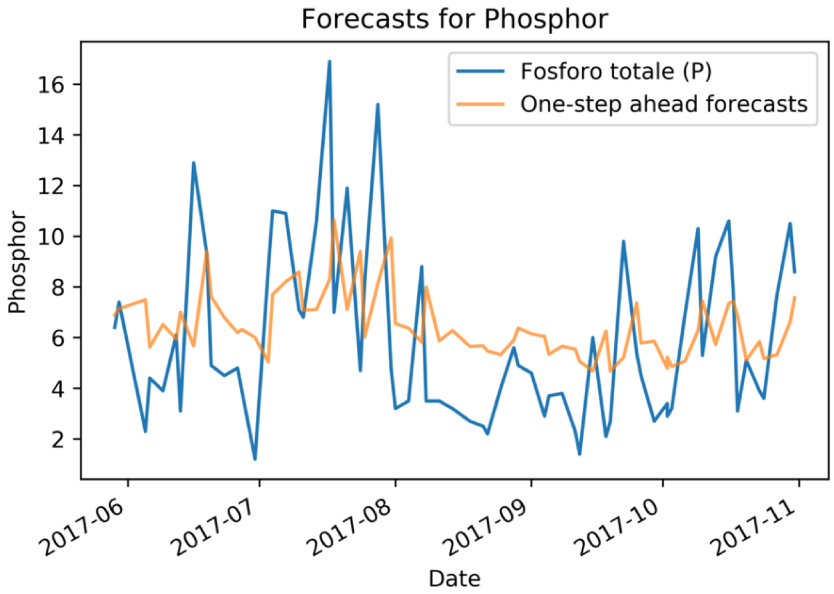


Figure 4.48: Forecasts for SARIMA model found for phosphor

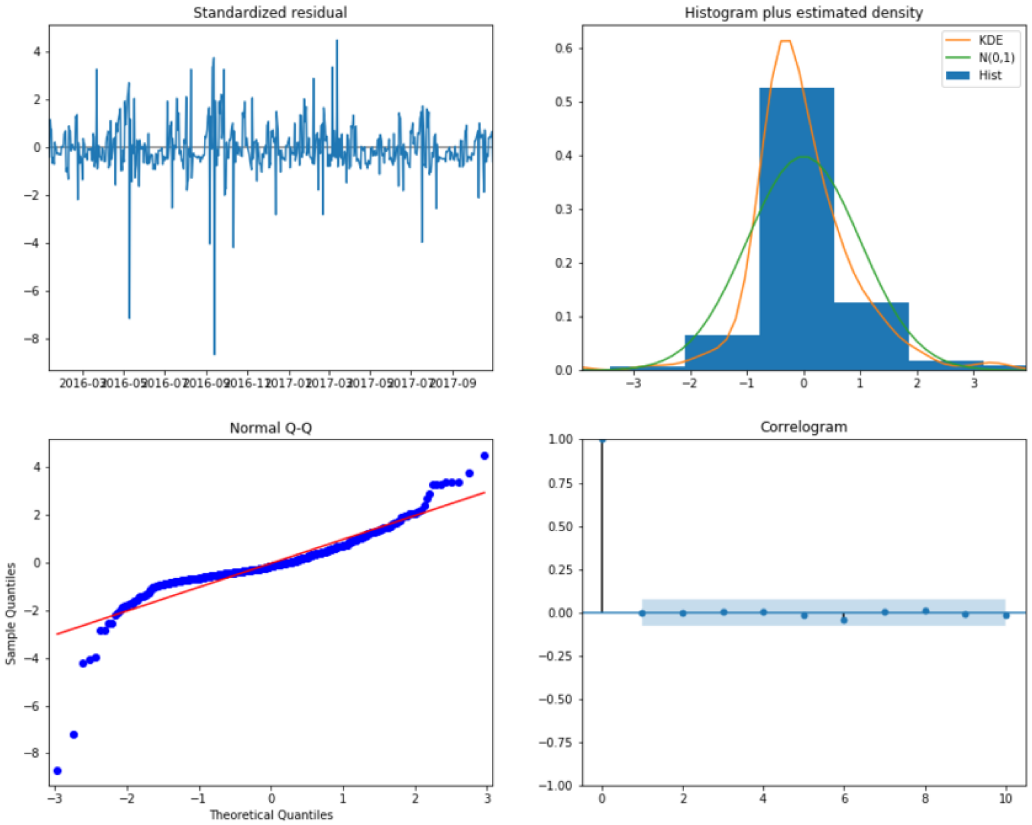


Figure 4.49: Diagnostics for SARIMA model found for phosphor resampled

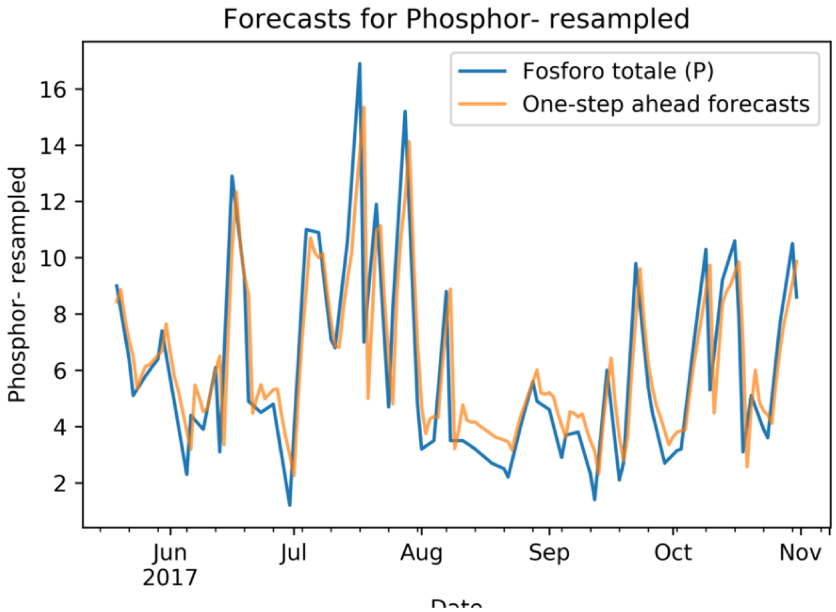


Figure 4.50: Forecasts for SARIMA model found for phosphor resampled

4.6.5 Sulphates

Chemical considerations. Results reported by chemical analysis simply show that sulphates are always found to be below legal threshold, with an unique peak during September 2016. No more indications or interpretations are given.

Time series Exploratory Data Analysis. The diagnostic for sulphates shows several aspects which have been already found for chloride and phosphor, starting from the gradual cutoff of the tails of both the ACF and the PACF. In this case, however, the histogram does not appear to be skewed.

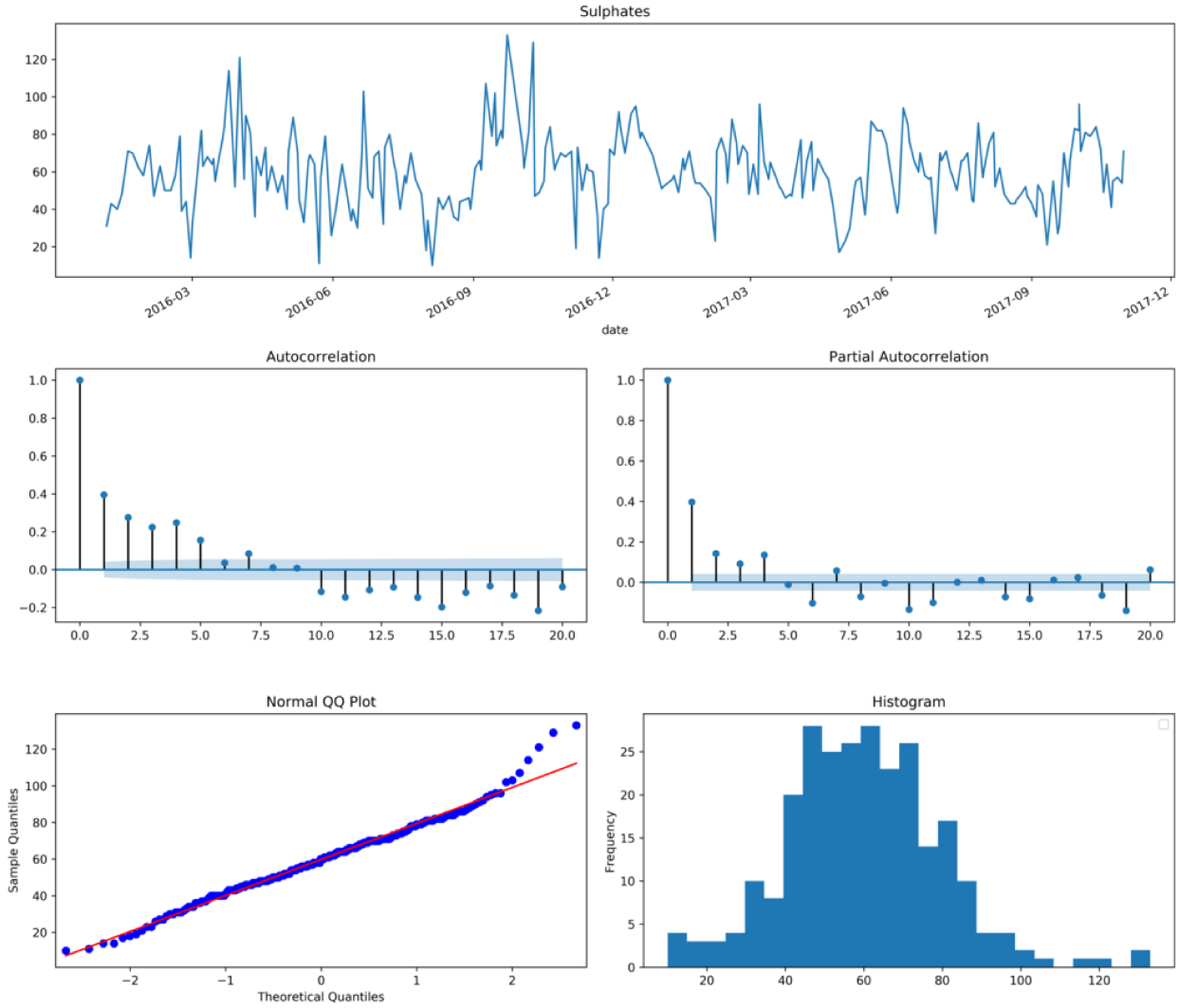


Figure 4.51: Analysis of sulphates for Vimercate Wastewater Treatment Plant

STL decomposition again does not highlight a global trend within data.

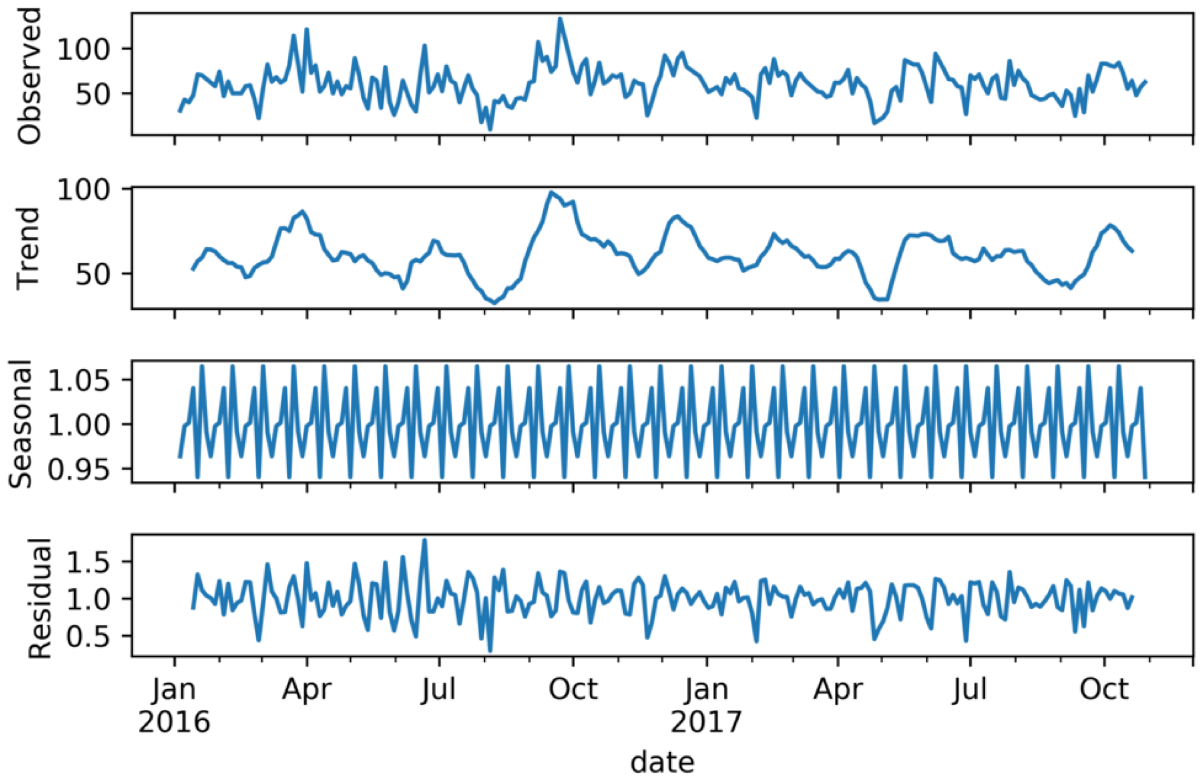


Figure 4.52: STL decomposition for sulphates

SARIMA modeling. Parameters for the best SARIMA model found for sulphates data are reported in table 4.52.

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
4	1	1	0	0	0	2240.54	219.44

Table 4.52: Parameters found for best SARIMA model on sulphates.

It should be remarked that, in this case, the value for the order of the MA component is found to be inferior than the one that should be expected observing the PACF plot. However, residuals appear to be normally distributed and uncorrelated.

Predictions confirm the usual pattern: the SARIMA model is able to follow the dynamic of the process, but not to capture sudden variations within data.

SARIMA modeling on resampled time series. Parameters for the best SARIMA model found for the resampled version of sulphates are reported in table 4.53.

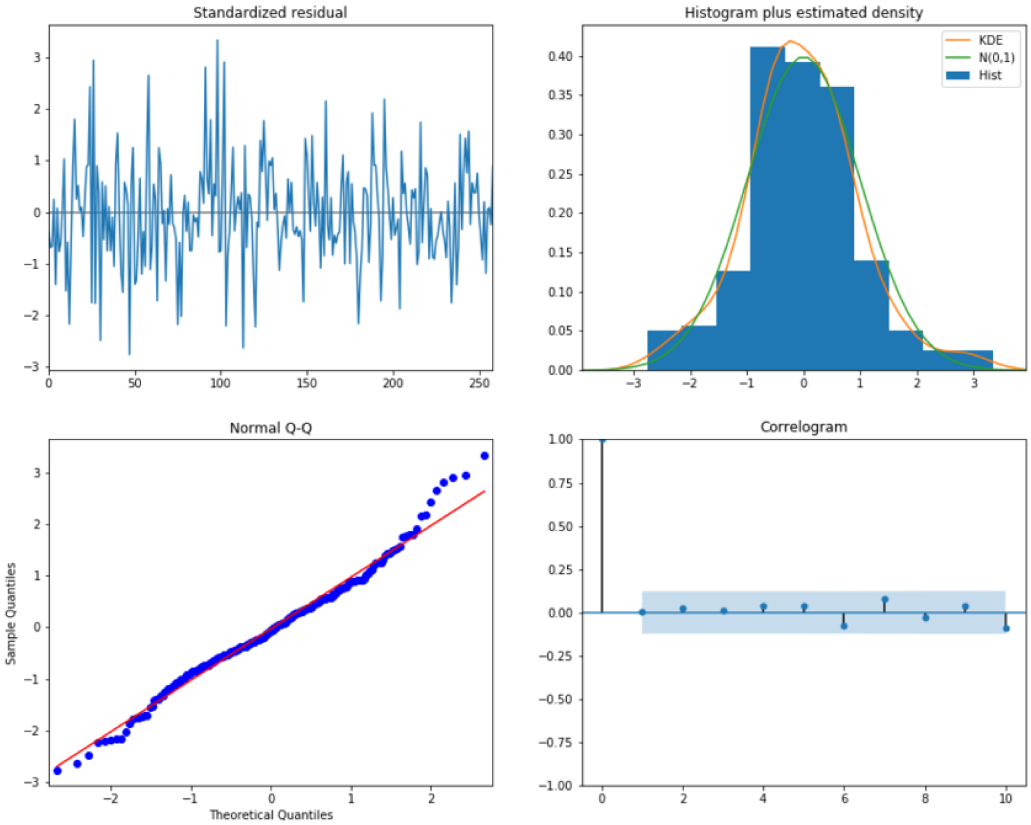


Figure 4.53: Diagnostics for SARIMA model found for sulphates

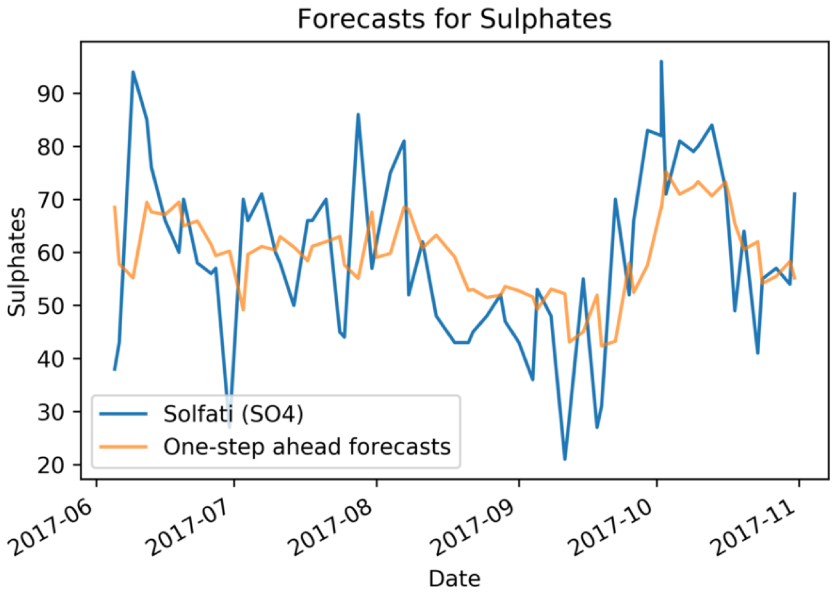


Figure 4.54: Forecasts for SARIMA model found for sulphates

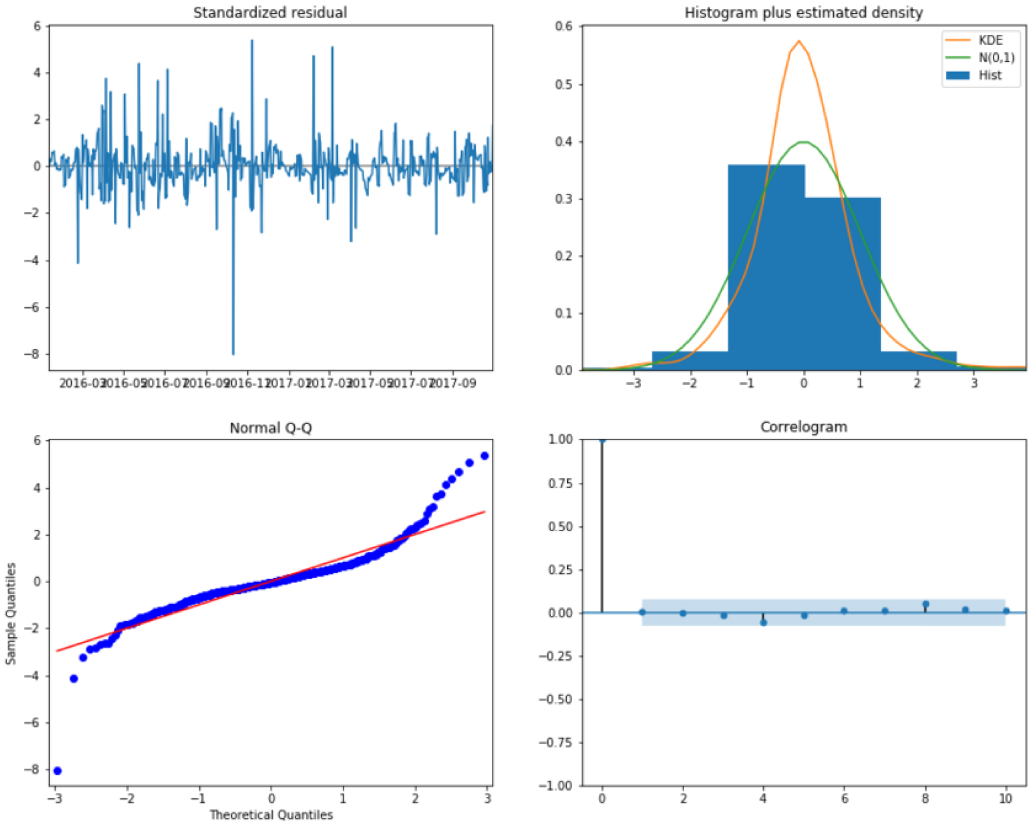


Figure 4.55: Diagnostics for SARIMA model found for sulphates resampled

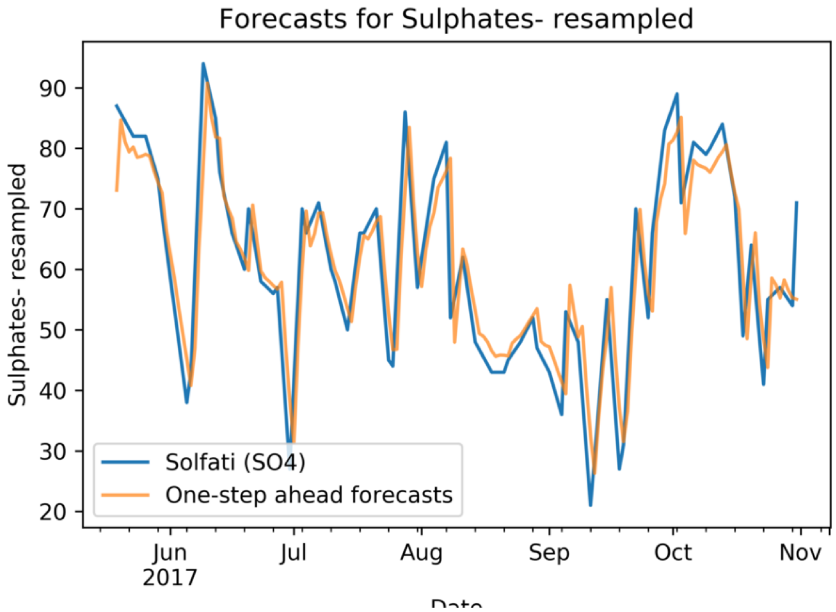


Figure 4.56: Forecasts for SARIMA model found for sulphates resampled

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
1	1	4	0	0	0	4807.96	41

Table 4.53: Parameters found for best SARIMA model on sulphates resampled.

Diagnostics for resampled time series highlight the presence of a kurtosis effect on the distribution of residuals. However, these do not appear to be correlated.

As expected, results are considerably better than the ones achieved by the SARIMA model for the original time series.

Remarks. No particular remarks, apart from the one previously found, can be made on this time series.

4.6.6 Suspended solids

Chemical considerations. The value for Total Suspended Solids fluctuates around the legal threshold of 200 mg/l, and is found to be almost always above this limit. The need for an extended analysis, which highlights the differences which can be found between the *volatile* suspended solids (that is, the part of suspended solids which is effectively into the water matrix) and the *sedimented* suspended solids, would be desirable as a future work.

Time series Exploratory Data Analysis. The time series shows a behavior which resembles the one followed by the phosphor, with skewed data, as shown by the histogram. Both ACF and PACF plots gradually tails off, with spikes at higher lags; the normal Q-Q plot resemble a normal distribution, with a considerable deviation on both the borders.

Again, STL does not suggest any global trend.

SARIMA modeling. Parameters for the best SARIMA model found for suspended solids data are reported in table 4.54.

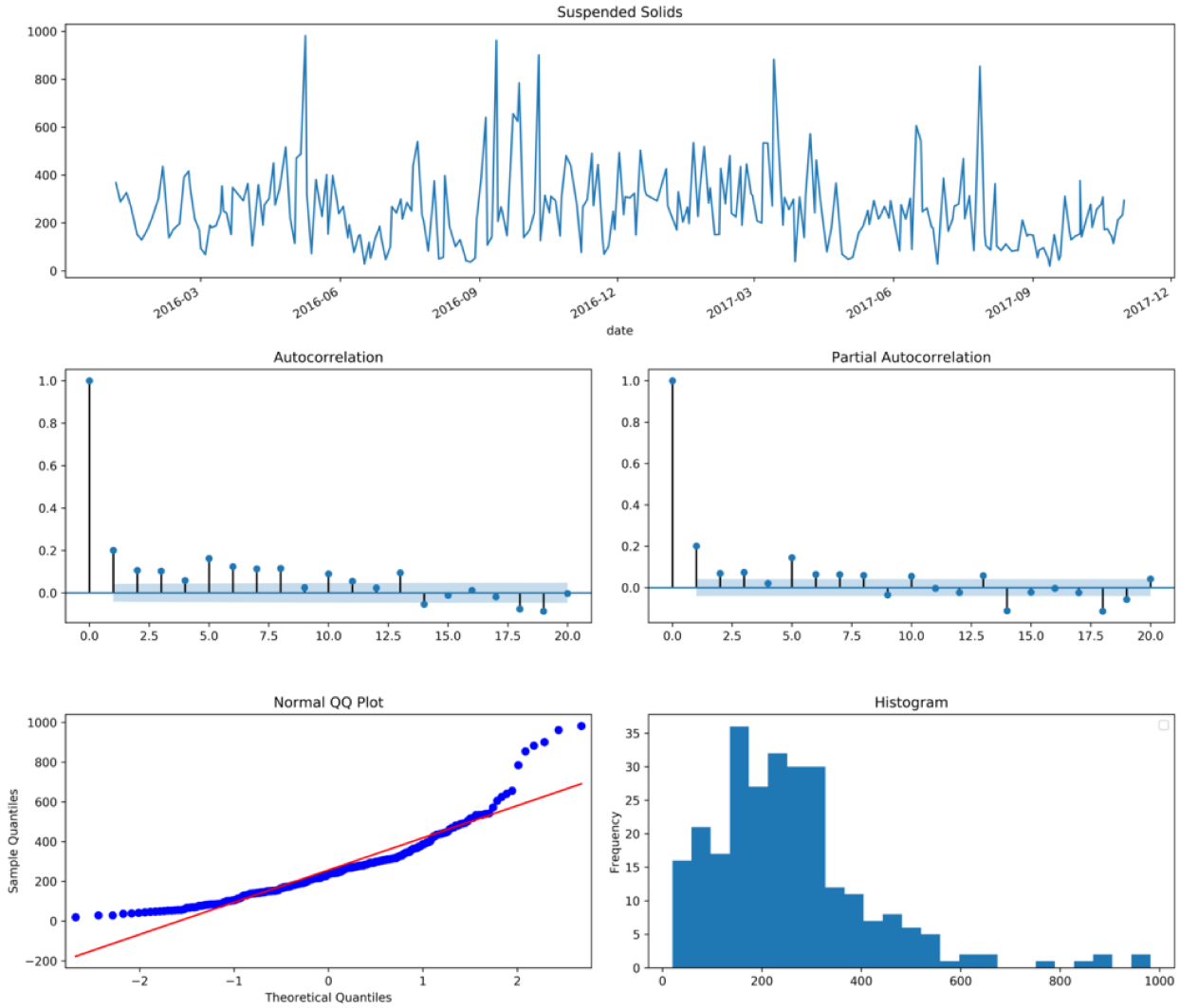


Figure 4.57: Analysis of suspended solids for Vimercate Wastewater Treatment Plant

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
0	1	4	0	0	0	3426.79	19473.13

Table 4.54: Parameters found for best SARIMA model on total suspended solids.

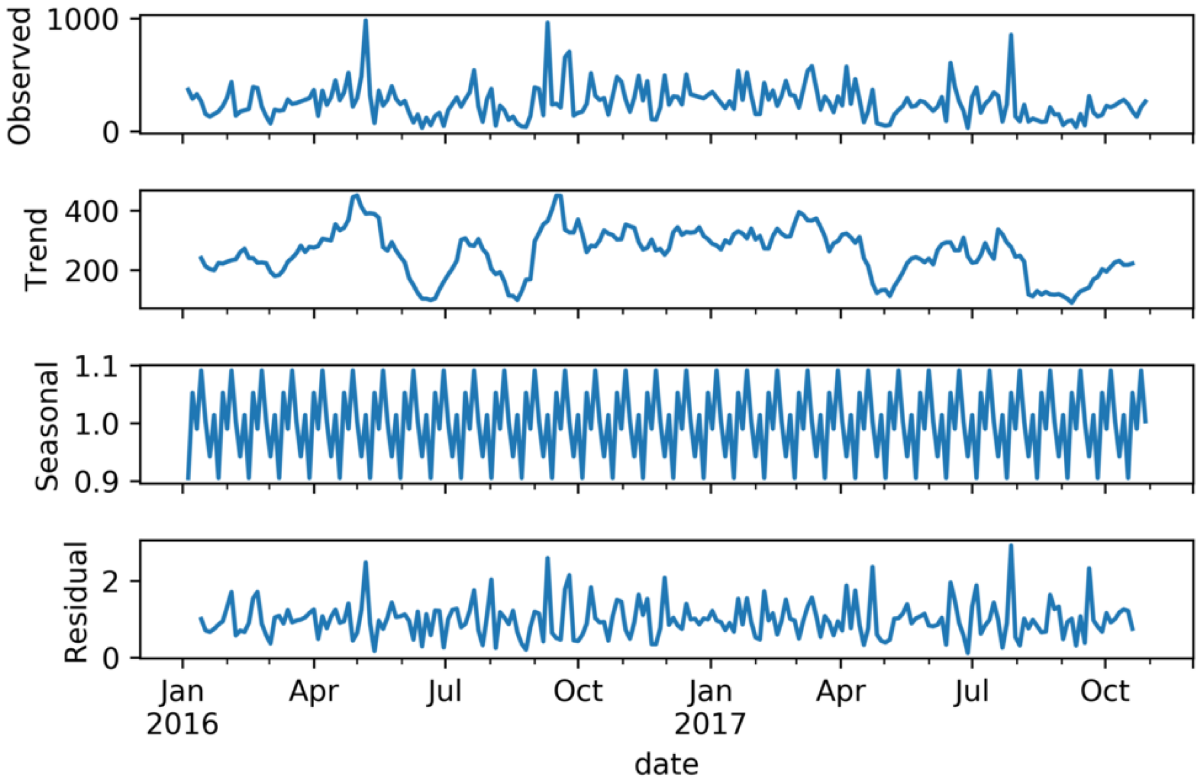


Figure 4.58: STL decomposition for suspended solids

There is nothing to remark on residuals found for the best SARIMA model on these data: these appear to be normally distributed and uncorrelated.

Predictions confirm that the model is not capable of properly characterize rapid changes in data.

SARIMA modeling on resampled time series. Parameters for the best SARIMA model found for the resampled version of COD are reported in table 4.55.

AR (p)	I (d)	MA (q)	sAR (P)	sI (D)	sMA (Q)	AIC	MSE forecasting
1	1	4	0	0	0	7822.15	4679.22

Table 4.55: Parameters found for best SARIMA model on suspended solids resampled.

Diagnostics show a kurtosis phenomena on residuals, if compared to results achieved on the original time series. Residuals, however, do not appear to be correlated.

As expected, forecasts of this model are capable to capture the quick variations within data.

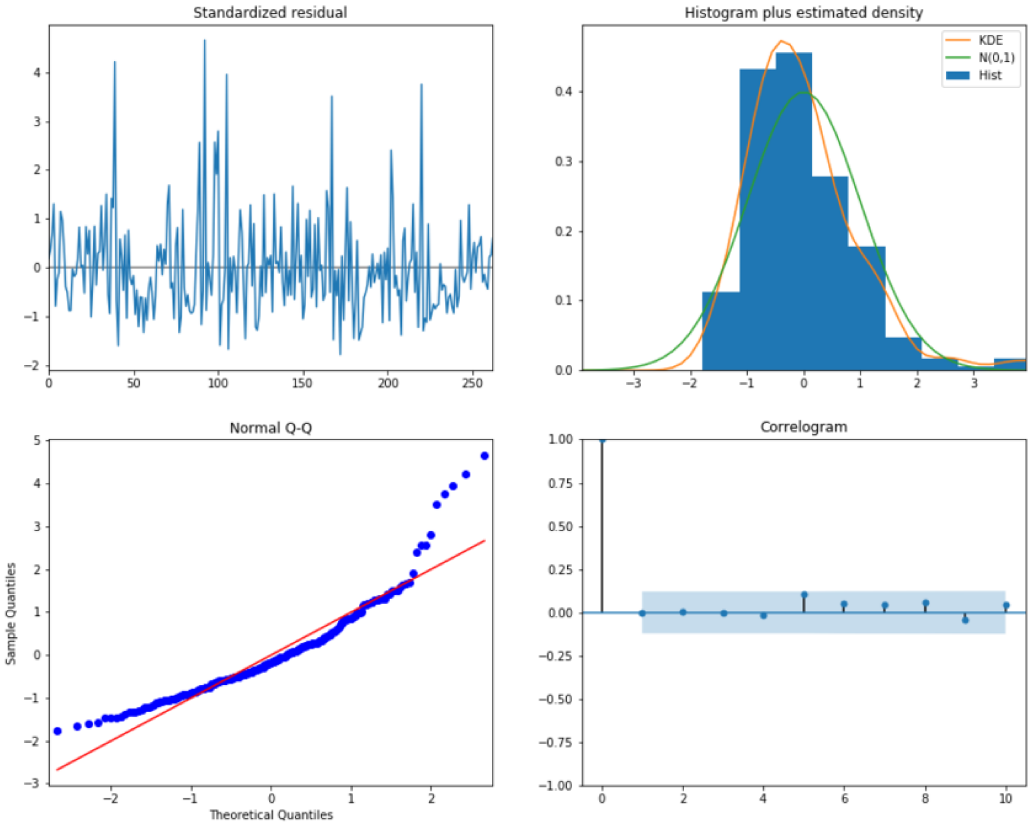


Figure 4.59: Diagnostics for SARIMA model found for suspended solids

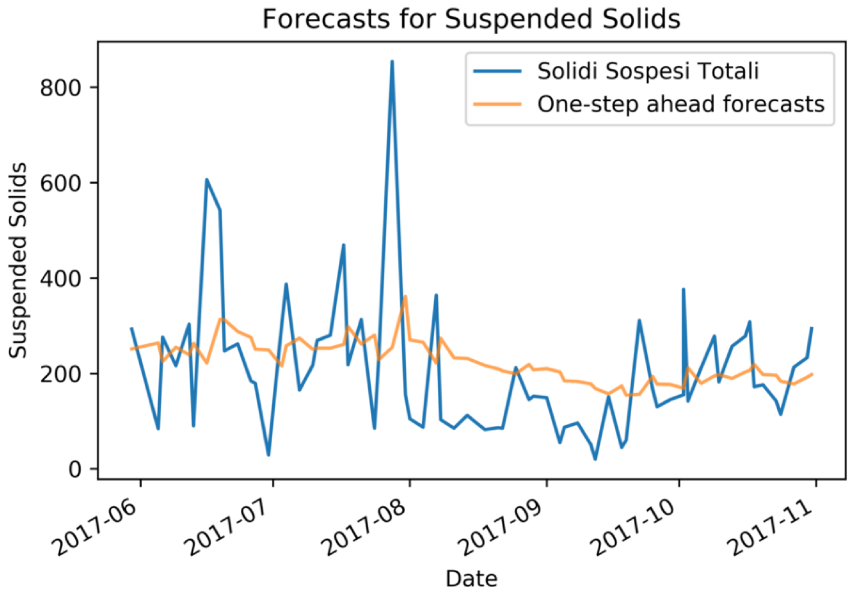


Figure 4.60: Forecasts for SARIMA model found for suspended solids

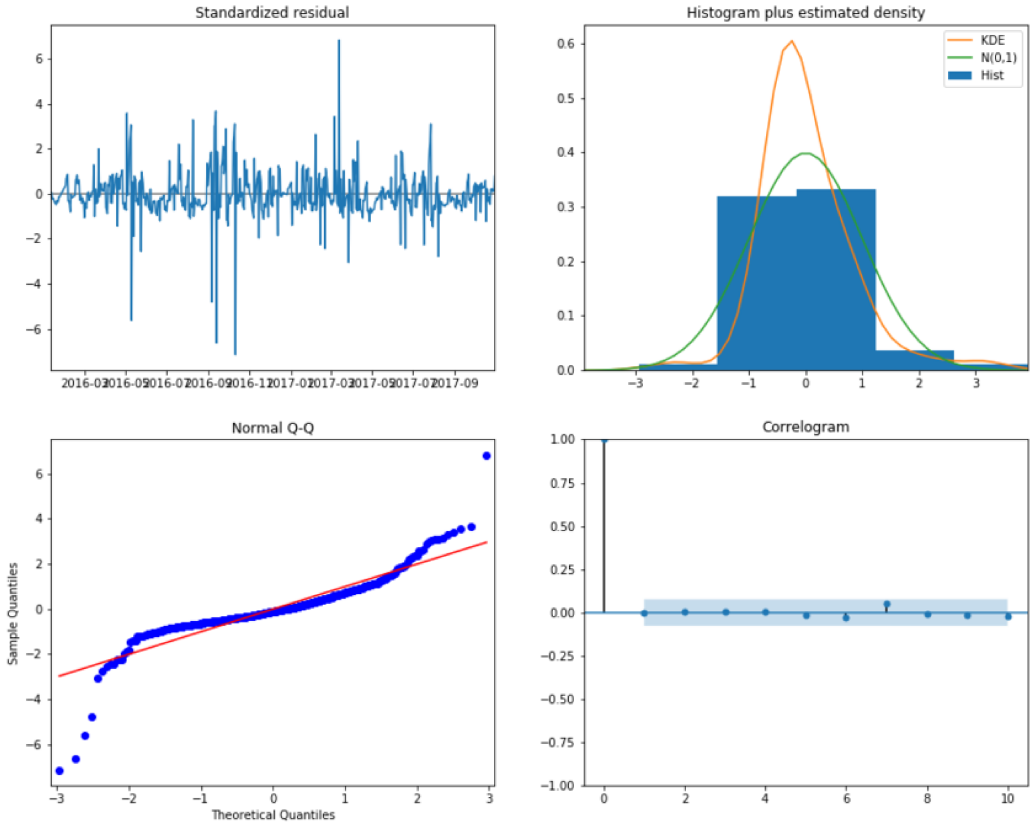


Figure 4.61: Diagnostics for SARIMA model found for suspended solids resampled

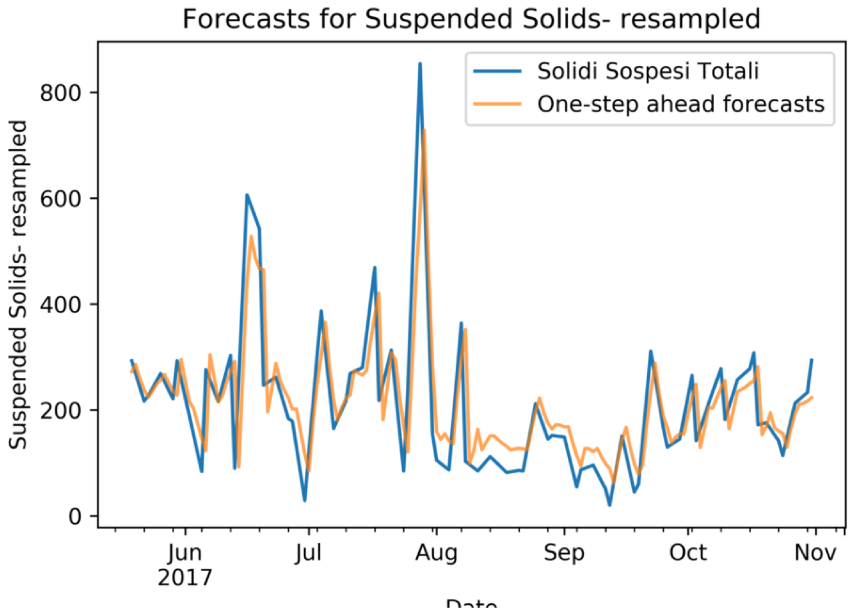


Figure 4.62: Forecasts for SARIMA model found for suspended solids resampled

Remarks. No particular remarks, apart from the one previously found, can be made on this time series.

4.6.7 Discussion

Analysis on time series contained in the IRSA Wastewater Dataset show a common problem: that is, the sampling campaign has not been properly designed.

The main design issue is that *data are not sampled on a regular basis*. By establishing a proper sampling methodology, with samples regularly acquired on daily (or even weekly) basis, this issue can be overcome, and modeling methods should be able to capture the intrinsic characteristics of underlying phenomena, as the behavior of SARIMA models show on oversampled time series.

Another issue lies in data themselves. As often happens in these case, the more data, the better: by taking a quick review on the literature, it is clear that proper modeling techniques require many years of continuous acquisition. Data coming from IRSA Wastewater do not show neither seasonal effects, nor trends, which is unexpected for environmental time series: seasonal effects, related to urban and industrial activity cycles, should be evident, as an overall trend which indicate whether pollution is increasing or the measures adopted to contrast pollution are being effective. Furthermore, instruments selected to gather data should be properly selected, as it has been highlighted in section 4.6.1 the negative impact of an excessively high detection threshold.

As such, future analysis should adhere to the following protocol:

- *regular acquisition*: samples should be taken once per day, if possible, or in any case on a regular basis;
- *dense acquisition*: apart from being regular, the time period between two consecutive acquisition should be as low as possible;

- *proper instrumentation*: when the acquisition campaign is being designed, expected values for the parameters should be assessed, and instruments with a proper resolution and detection threshold should be chosen;

- *long, continuous acquisition*: the acquisition campaign should be designed to have a long temporal horizon (at least five years), to capture trends and seasonal effects.

4.7 Multivariate analysis with complex networks

Despite not being one of the most relevant from the *quantitative* point of view, this section describes one of the most important results which have achieved during this work, that is, a methodology for exploring the interactions between the responses of the sensors within an e-nose. Specifically, the goal of this method is to verify if, through a multivariate analysis, it is possible to define an optimal configuration for the sensors array.

It is clear that a gas sensor array can be framed as a complex system. There are complex interactions between each sensor in the array, which can dynamically change over time. Furthermore, responses of each sensor can be correlated: as a consequence, there may be redundancies within the array, or, by analyzing the correlation map, one may infer the substances found within the array with enhanced precision.

Hence, some of the ideas on which this work is based can be directly borrowed from neurosciences, which is a field where complex networks are widely employed to model the complex interactions between various areas of the brain [152]. Following this approach, a complex network is built starting from bot the sensors within the e-nose (which are the nodes of the network) and the correlation of the signals acquired by each possible couple of sensors (which are the edges of such network).

Let us remark again that, unlike the univariate modeling presented in section 4.6, this approach is inherently *multivariate*, and allows to simultaneously consider several conditioning factors that could affect the sensing of chemical compounds, such as environmental condition and

technical specifications of the devices.

The main purpose of this work, which, at the time this thesis is being written, has been submitted for review [153], is to perform an exploratory analysis which investigates a method for identifying a minimum set of both configuration parameters and flow conditions that allows to properly discriminate responses of the e-nose to different compounds.

4.7.1 Mathematical description

Let us now briefly describe the mathematical approach used in the modeling.

First, recalling chapter 2 the e-nose has been modeled as a complex network $G = (V, E)$, where the set of nodes $V = v_1, \ldots, v_n$ represent the sensors within the array, and edges $E = e_{12}, \ldots, e_{nn}$ are defined according to the correlation between each couple of sensors. In this case, since the dataset described in 4.3.2 is used, n is set to 72, while the number of edge is $\frac{(n(n-1))}{2}$, as the network is supposed to be dense.

As already stated, edges are defined in a way similar to the one adopted to define functional connectivity in brain network. Therefore, the edge e_{ij} between nodes v_i and v_j is defined using the correlation coefficients between time series y_{t_i} and y_{t_j} acquired by nodes v_i and v_j, respectively. Formally:

$$e_{ij} = \tau \left(y_{t_i}, y_{t_j} \right) \tag{4.7}$$

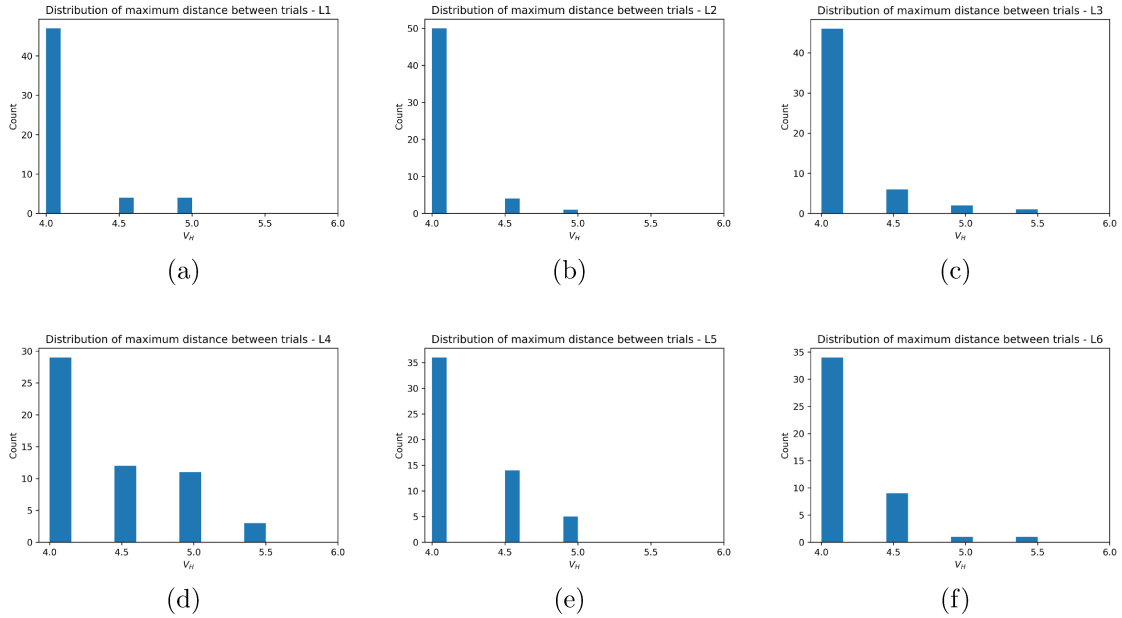
In 4.7, τ is the Kendall correlation coefficient. We choose to use τ over other correlation coefficients (e.g., Pearson r or Spearman ρ), because we could not suppose neither linearity nor monotonicity between y_{t_i} and y_{t_j}.

It is intuitive to say that the configuration of the network G is related to the overall response of the e-nose to a specific compound. As a consequence, if the responses to two different substances are different, the idea is that these substances may be easily discriminated by the respective *signatures*.

The evaluation of the discriminative capabilities of such approach are evaluated as follows.

1. First, the number of degrees of freedom within data is identified. In this case, there are three degrees of freedom; the first two are determined by the conditioning parameters, that is, the values for V_h and S (cfr. section 4.3.2). The third one is given by the layer l at which the measurement is performed. In the experiments, a large part of possible cases has been covered, using a set of networks G_{vsl}, with $v \in V_h, s \in S, l \in L$. Each network models the response of the signal to each chemical compound as values for voltage heater, fan speed and level are fixed.

2. Once network configurations are determined, a signature is obtained for each layer for each compound, represented by an edge set E. Then, the distance among these signature is evaluated by using cosine distance. Following this intuition, high values of cosine distance allows to easily discern between the overall responses of the e-nose to each couple of chemical substances.

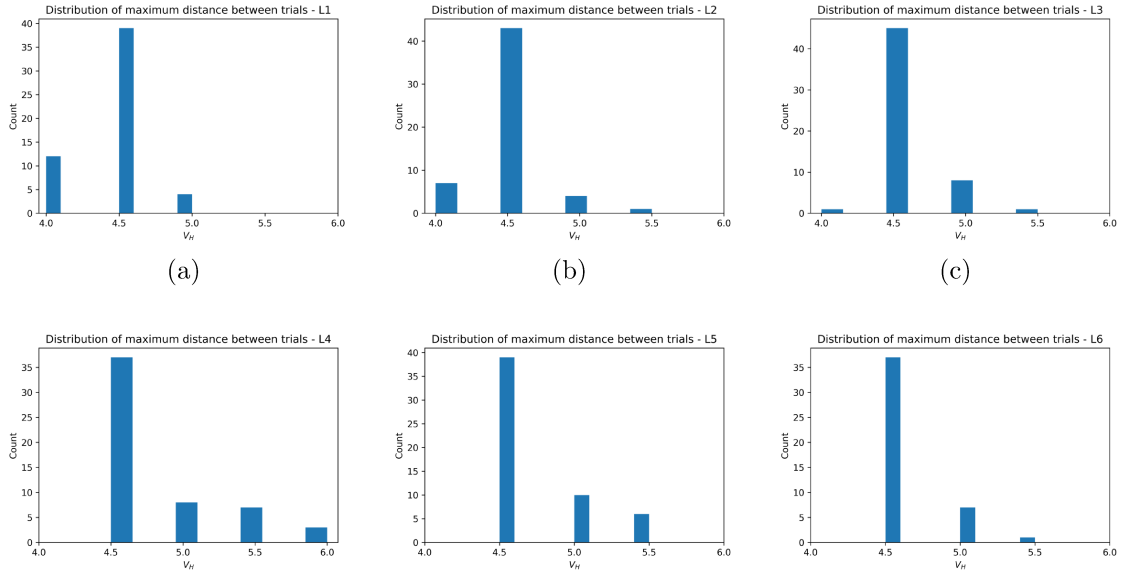


(a) (b) (c)

(d) (e) (f)

Figure 4.63: Count of maximum distances varying V_h and l with fixed $S = 0.10 m/s$.

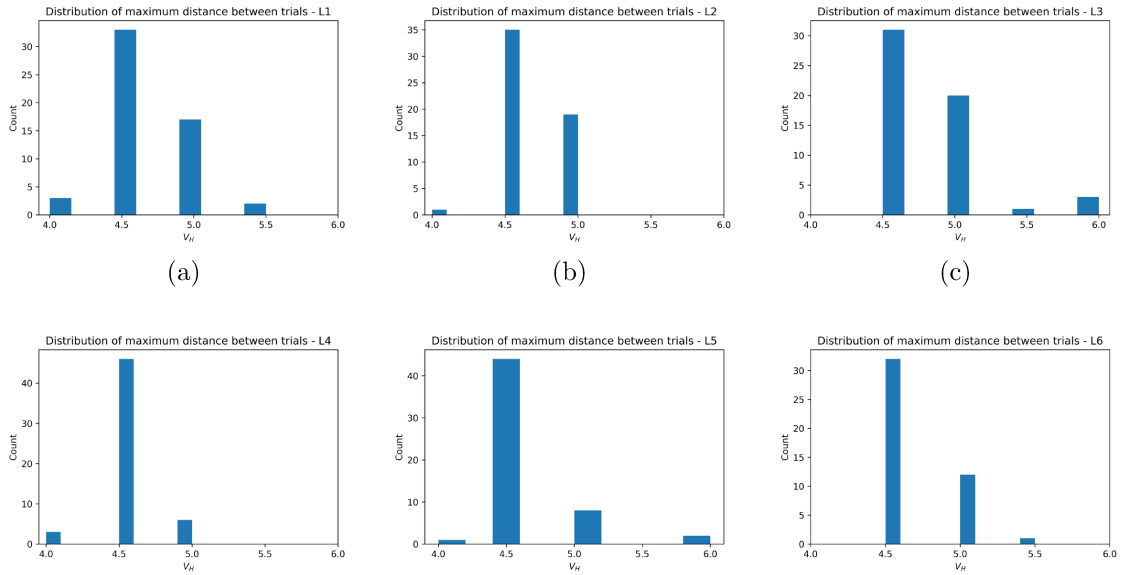
Figures 4.63, 4.64 and 4.65 show which voltage value accounts for the maximum distances between trials when V_h and l vary, with three fixed values for S, that is, $S = 0.10\ m/s$, $S = 0.21\ m/s$ and $S = 0.34\ m/s$.



(a) (b) (c)

(d) (e) (f)

Figure 4.64: Count of maximum distances varying V_h and l with fixed $S = 0.21m/s$.



(a) (b) (c)

(d) (e) (f)

Figure 4.65: Count of maximum distances varying V_h and l with fixed $S = 0.34m/s$.

Interestingly, it can be seen that the behavior of the system when these parameters vary is consistent. In fact, for a fixed value of $S = 0.10m/s$, the maximum distances between the signatures of the e-nose appear to be concentrated when the input voltage is $V_h = 4.0V$, while for $S = 0.21m/s$ and $S = 0.34m/s$ these appear to be concentrated to $V_h = 4.5V$. Obviously, these information can be used to achieve the maximum discriminative power for the instrument.

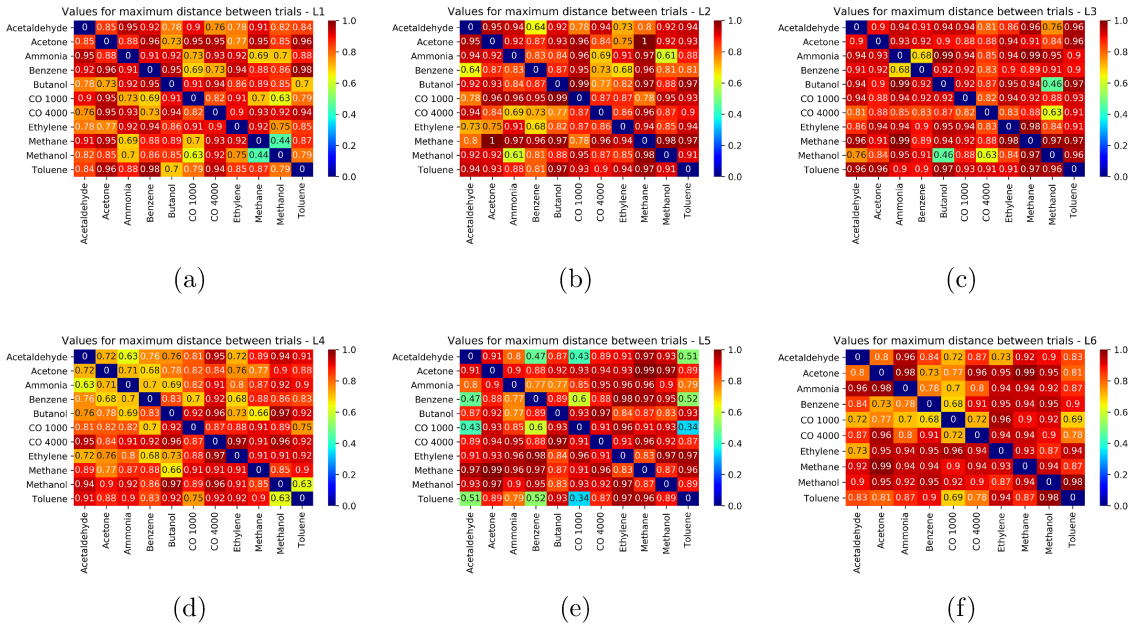


Figure 4.66: Maximum distances for the most discriminative V_h varying L with fixed $S = 0.10m/s$.

Let us now focus of figures 4.66, 4.67 and 4.68. From these, it is possible to observe values for the cosine distance when V_h is fixed at the values of $4.0V$ for $S = 0.10m/s$ and $4.5V$ otherwise. Visually, the heat maps with warmer colors are more discriminative, as the distance values D are on average higher.

These findings are better summarized in Table 4.56 which shows the average cosine distance values for each of the three fan speeds for the most discriminative V_h and varying L.

Fan speed	Optimal V_h	L_1	L_2	L_3	L_4	L_5	L_6
0.10 m/s	4.0	0.85	0.88	0.89	0.83	0.86	0.87
0.21 m/s	4.5	0.71	0.73	0.70	0.72	0.78	0.74
0.34 m/s	4.5	0.57	0.61	0.59	0.68	0.72	0.73

Table 4.56: Average cosine distance value for the most discriminative V_h varying L.

These results are confirmed by the ones achieved in [96]. Specifically, S is found to be the most

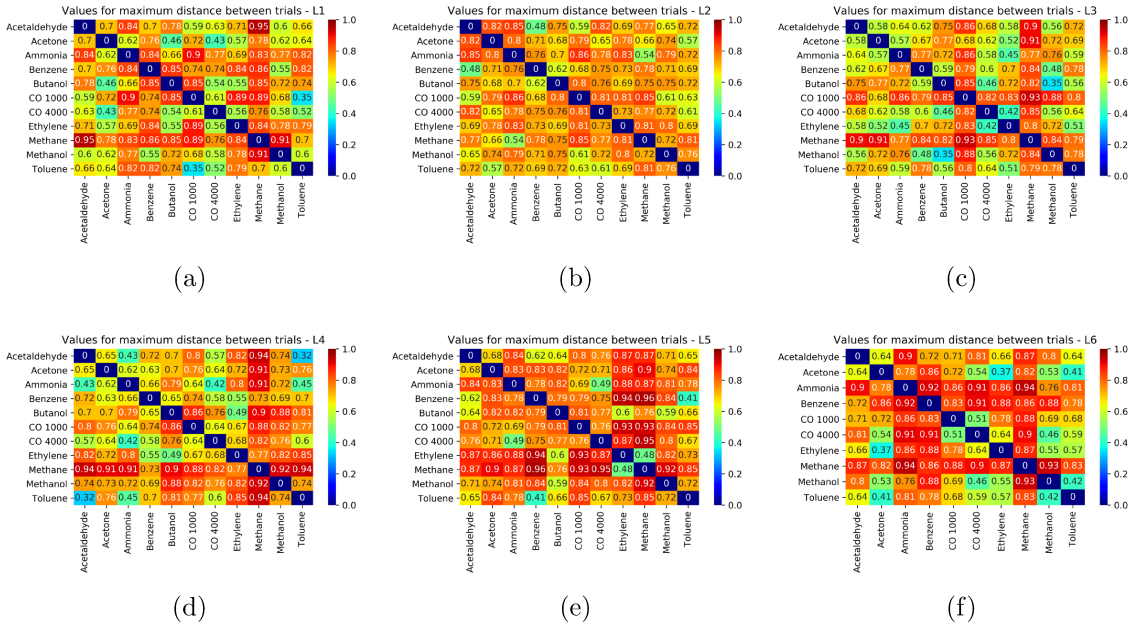


Figure 4.67: Maximum distances for the most discriminative V_h varying L with fixed $S = 0.21m/s$.

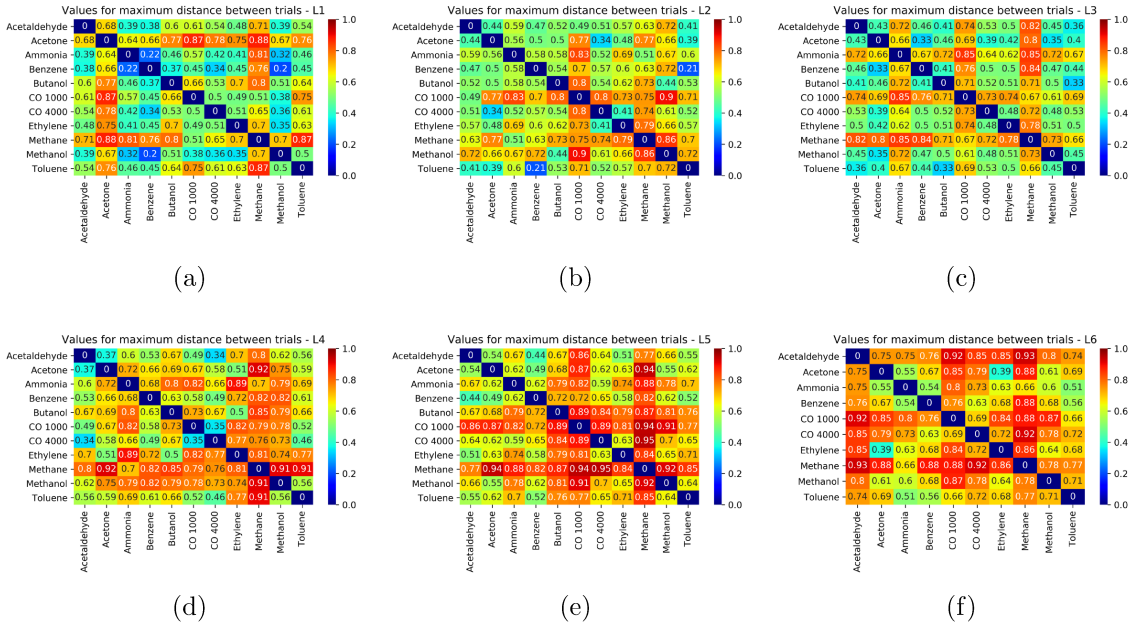


Figure 4.68: Maximum distances for the most discriminative V_h varying L with fixed $S = 0.34m/s$.

relevant conditioning parameter, but several more hints on the relevance of L and V_h are given.

First, L seems to be relevant for higher values of S due to the fact that initial layers are mostly affected by turbulence aspects already described in [96]. The bar plots in figures 4.63, 4.64 and 4.65 also shows that the value of the input voltage to the heater is particularly relevant, and depends on the specific value of the fan speed. Finally, it can be noted that the maximum discrimination among the compounds is achieved at low speed, as the influence of turbulence on the system is lower.

The implications of these results, which are backed up by the findings of the authors of the dataset themselves, is clear: a new, versatile and data driven method can be used to automatically select the most appropriate value for conditioning parameters to apply to an e-nose to achieve the maximal selectivity, and, therefore, enhance its performance in the specific application scenario. However, it must be underlined that this method cannot give a remedy to a poor design of the experimental settings, and must be therefore properly used in a well-defined evaluation pipeline.

4.8 EnvLab

In the last section of this chapter, a brief overview on *EnvLab*, the Python library developed during this work, is given.

EnvLab has been developed for mainly two purposes. The first one is to allow the repeatability of the experiments performed in this work, while the second one is to create a basis for the use of the proposed processing pipelines to other use cases, with a particular focus on environmental-related scenarios. Therefore, EnvLab has been developed and released as a free and open source software.

The library, which, at the time this thesis is being written is in a preliminary alpha stage, has been built on to of other well-known Python libraries, commonly used for machine learning and data analysis, such as Pandas, Numpy, Scikit-Learn and TensorFlow.

It is important to underline that EnvLab is not currently intended to act as a replacement for such libraries, whose breath is obviously (and intentionally) wider, as they can be used as general-purpose libraries. Instead, EnvLab is strictly focuse don the pipelines defined and implemented in this work, and therefore does not currently offer an API which can extend these pipelines.

However, as this thesis work finishes, more effort will be put in the development of the library, and, starting from the takeaway given by other related projects [154], the development of a complete, coherent and easy-to-use API, which can be used to create new working pipelines that can be adapted to several environmental monitoring scenarios, will be achieved.

EnvLab is currently available at the following address: https://github.com/anhelus/env_lab.

Chapter 5

Conclusion

5.1 A perspective on future works

It is clear that this work is still a first step towards the definition of a complete working pipeline for environmental data analysis.

More experiments are needed: the processing pipelines should be further refined, eventually by considering wider, and more significant, experiments, and innovative approaches, such as the ones which use transfer learning and complex networks can be improved by extended experiments.

Furthermore, EnvLab will be further developed in the future, hopefully with the contribution of the open source community, to a stable (and therefore well tested, documented and accepted) release, acting as a basis for environmental data analysis.

5.2 Final thoughts

It has not been easy to summarise the contribution of this work. In fact, its spectrum has been wide enough to 'touch' several disciplines related to signal processing and machine learning;

yet, there is a single, and extremely important, take-home message, which, hopefully, will be taken by the reader.

This whole work is a *test*. A *first*, and hopefully *meaningful* step towards the definition of a 'working pipeline' for environmental analysis.

As a matter of fact, this work does not offer innovative methods for machine learning and data analysis in the environmental field: however, it should be clear that the differences between *theory* and *real world* have *extremely deep implications*, and, even a rigorous processing procedure cannot deal with a bad experimental design.

A real *multidisciplinarity* is needed. All the involved parties, each one with its own expertise, should interact in the data processing pipelin. The chemical expert should point out the expected results, which should be understood by both the hardware maker, to properly choose the hardware to use, and the data scientist, who should therefore adopt the most well-suited techniques for the analysis. But also the other parties, the ones not directly involved, must do their part to *disseminate* the results and avoid bureaucratic issues.

And a continuous feedback is needed, in the search for a virtuous *expertise backpropagation* which, hopefully, will help us all to better understand the complex system we all are embedded into. And, perhaps, to save us from our ignorance.

Appendix A

Other works

Even if the main focus of this thesis was on environmental monitoring, other works have been realized during the PhD. These have lead, either directly or indirectly, to the development of the ideas and the scientific background on which the methods shown in this thesis have been built.

In this appendix, a brief overview of such works will be given.

First, in [159], an approach to person re-identification, a well-known problem in video surveillance system, has been proposed. This approach tried to compare re-identification approaches using an innovative methodology, based on a score related to the complexity in the identification of each individual captured by the video surveillance system. The work was extended and published as a book chapter [160]. The main contribution to this thesis of these work lied in the exploration of the capabilities of convolutional neural networks, with transfer learning which lead the intuitions used in the classification of data acquired by the VPeNs.

In [161], a real-time system to identify the noise source within cluttered environments was proposed. This system used an array with three microphones to achieve state-of-the-art performance. The main contribution to this thesis of this work lied in a better knowledge of the problems related to signal processing, which helped during the definition of the protocol used for opportunistic sensing.

Two other works, specifically [162] and [163], were used as a basis for the knowledge related to IoT and signal processing which was then used in the evaluation and contribution to the development of the VPeN.

Finally, one of the most important work has been the one defined with Angela Lombardi and Prof. Guaragnella in [164]. This work lead to the definition of a resilience-based measure for complex network, which will be used as a basis for further developments of the most innovative part of the thesis (that is, the multivariate analysis of gas sensor arrays through complex network).

Bibliography

[1] https://climate.nasa.gov/

[2] Le Qur, C., Andrew, R. M., Friedlingstein, P., Sitch, S., Pongratz, J., Manning, A. C., ... and Boden, T. A. (2017). Global carbon budget 2017. Earth System Science Data Discussions, 1-79.

[3] Norton, A., and Silberger, A. J. (1987). Parallelization and performance analysis of the Cooley-Tukey FFT algorithm for shared-memory architectures. IEEE Transactions on Computers, 36(5), 581-591.

[4] Tukey, J. W. (1977). Exploratory data analysis (Vol. 2).

[5] Dundar, M., Krishnapuram, B., Bi, J., and Rao, R. B. (2007, January). Learning Classifiers When the Training Data Is Not IID. In IJCAI (pp. 756-761).

[6] Newman, M. E., and Girvan, M. (2004). Finding and evaluating community structure in networks. Physical review E, 69(2), 026113.

[7] Russell, S. J., and Norvig, P. (2016). Artificial intelligence: a modern approach. Malaysia; Pearson Education Limited,.

[8] Hastie, T., Tibshirani, R., and Friedman, J. (2009). Unsupervised learning. In The elements of statistical learning (pp. 485-585). Springer, New York, NY.

[9] Szegedy, C., Zaremba, W., Sutskever, I., Bruna, J., Erhan, D., Goodfellow, I., and Fergus, R. (2013). Intriguing properties of neural networks. arXiv preprint arXiv:1312.6199.

[10] Harris, D., and Harris, S. (2010). Digital design and computer architecture. Morgan Kaufmann.

[11] Cardellicchio, A., Dentamaro, G., Di Lecce, V., Guaragnella, C., and Rizzi, M. (2016, June). An opportunistic sensor network approach to wide area environmental sensing. In Environmental, Energy, and Structural Monitoring Systems (EESMS), 2016 IEEE Workshop on (pp. 1-6). IEEE.

[12] Deng, J., Dong, W., Socher, R., Li, L. J., Li, K., and Fei-Fei, L. (2009, June). Imagenet: A large-scale hierarchical image database. In Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on (pp. 248-255). IEEE.

[13] Krizhevsky, A., Sutskever, I., and Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In Advances in neural information processing systems (pp. 1097-1105).

[14] Hubel, D. H., and Wiesel, T. N. (1968). Receptive fields and functional architecture of monkey striate cortex. The Journal of physiology, 195(1), 215-243.

[15] Euler, L. (1736). Solutio problematis ad geometriam situs pertinensis. Comm. Acad. Sci. Imper. Petropol., 8, 128-140.

[16] Boccaletti, S., Latora, V., Moreno, Y., Chavez, M., and Hwang, D. U. (2006). Complex networks: Structure and dynamics. Physics reports, 424(4-5), 175-308.

[17] Brockwell, P. J., and Davis, R. A. (2016). Introduction to time series and forecasting. springer.

[18] Westwick, D. T., and Kearney, R. E. (2003). Identification of nonlinear physiological systems (Vol. 7). John Wiley and Sons.

[19] Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. (2015). Time series analysis: forecasting and control. John Wiley and Sons.

[20] Cleveland, R. B., Cleveland, W. S., McRae, J. E., and Terpenning, I. (1990). STL: A Seasonal-Trend Decomposition. Journal of Official Statistics, 6(1), 3-73.

[21] Dickey, D. A., and Fuller, W. A. (1979). Distribution of the estimators for autoregressive time series with a unit root. Journal of the American statistical association, 74(366a), 427-431.

[22] Schalkoff, R. J. (1997). Artificial neural networks (Vol. 1). New York: McGraw-Hill.

[23] Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. Psychological review, 65(6), 386.

[24] Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. Neural computation, 9(8), 1735-1780.

[25] Connor, J. T., Martin, R. D., and Atlas, L. E. (1994). Recurrent neural networks and robust time series prediction. IEEE transactions on neural networks, 5(2), 240-254.

[26] Bengio, Y., Simard, P., and Frasconi, P. (1994). Learning long-term dependencies with gradient descent is difficult. IEEE transactions on neural networks, 5(2), 157-166.

[27] Gers, F. A., and Schmidhuber, J. (2000, July). Recurrent nets that time and count. In ijcnn (p. 3189). IEEE.

[28] Cho, K., Van Merrinboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. (2014). Learning phrase representations using RNN encoder-decoder for statistical machine translation. arXiv preprint arXiv:1406.1078.

[29] Erdos, P., and Rnyi, A. (1960). On the evolution of random graphs. Publ. Math. Inst. Hung. Acad. Sci, 5(1), 17-60.

[30] Barabsi, A. L., and Bonabeau, E. (2003). Scale-free networks. Scientific american, 288(5), 60-69.

[31] Srivastava, A. K. (2003). Detection of volatile organic compounds (VOCs) using SnO2 gas-sensor array and artificial neural network. Sensors and Actuators B: Chemical, 96(1-2), 24-37.

[32] Capelli, L., Sironi, S., and Del Rosso, R. (2014). Electronic noses for environmental monitoring applications. Sensors, 14(11), 19979-20007.

[33] Distante, C., Leo, M., Siciliano, P., and Persaud, K. C. (2002). On the study of feature extraction methods for an electronic nose. Sensors and Actuators B: Chemical, 87(2), 274-288.

[34] Ehret, B., Safenreiter, K., Lorenz, F., and Biermann, J. (2011). A new feature extraction method for odour classification. Sensors and Actuators B: Chemical, 158(1), 75-88.

[35] Brezmes, J., Ferreras, B., Llobet, E., Vilanova, X., and Correig, X. (1997). Neural network based electronic nose for the classification of aromatic species. Analytica Chimica Acta, 348(1-3), 503-509.

[36] Gardner, J. W. (1991). Detection of vapours and odours from a multisensor array using pattern recognition Part 1. Principal component and cluster analysis. Sensors and Actuators B: Chemical, 4(1-2), 109-115.

[37] Zhang, Q., Xie, C., Zhang, S., Wang, A., Zhu, B., Wang, L., and Yang, Z. (2005). Identification and pattern recognition analysis of Chinese liquors by doped nano ZnO gas sensor array. Sensors and Actuators B: Chemical, 110(2), 370-376.

[38] Tomchenko, A. A., Harmer, G. P., Marquis, B. T., and Allen, J. W. (2003). Semiconducting metal oxide sensor array for the selective detection of combustion gases. Sensors and Actuators B: Chemical, 93(1-3), 126-134.

[39] Choi, N. J., Kwak, J. H., Lim, Y. T., Bahn, T. H., Yun, K. Y., Kim, J. C., ... and Lee, D. D. (2005). Classification of chemical warfare agents using thick film gas sensor array. Sensors and Actuators B: Chemical, 108(1-2), 298-304.

[40] Maaten, L. V. D., and Hinton, G. (2008). Visualizing data using t-SNE. Journal of machine learning research, 9(Nov), 2579-2605.

[41] Nicolas, J., Romain, A. C., Wiertz, V., Maternova, J., and Andr, P. (2000). Using the classification model of an electronic nose to assign unknown malodours to environmental

sources and to monitor them continuously. Sensors and Actuators B: Chemical, 69(3), 366-371.

[42] Carmel, L., Levy, S., Lancet, D., and Harel, D. (2003). A feature extraction method for chemical sensors in electronic noses. Sensors and Actuators B: Chemical, 93(1-3), 67-76.

[43] Aggarwal, C. C., Hinneburg, A., and Keim, D. A. (2001, January). On the surprising behavior of distance metrics in high dimensional space. In International conference on database theory (pp. 420-434). Springer, Berlin, Heidelberg.

[44] Martn, Y. G., Oliveros, M. C. C., Pavn, J. L. P., Pinto, C. G., and Cordero, B. M. (2001). Electronic nose based on metal oxide semiconductor sensors and pattern recognition techniques: characterisation of vegetable oils. Analytica Chimica Acta, 449(1-2), 69-80.

[45] Oliveros, M. C. C., Pavn, J. L. P., Pinto, C. G., Laespada, M. E. F., Cordero, B. M., and Forina, M. (2002). Electronic nose based on metal oxide semiconductor sensors as a fast alternative for the detection of adulteration of virgin olive oils. Analytica Chimica Acta, 459(2), 219-228.

[46] Sohn, J. H., Hudson, N., Gallagher, E., Dunlop, M., Zeller, L., and Atzeni, M. (2008). Implementation of an electronic nose for continuous odour monitoring in a poultry shed. Sensors and Actuators B: Chemical, 133(1), 60-69.

[47] Vergara, A., Vembu, S., Ayhan, T., Ryan, M. A., Homer, M. L., and Huerta, R. (2012). Chemical gas sensor drift compensation using classifier ensembles. Sensors and Actuators B: Chemical, 166, 320-329.

[48] Huerta, R., Vembu, S., Amig, J. M., Nowotny, T., and Elkan, C. (2012). Inhibition in multiclass classification. Neural computation, 24(9), 2473-2507.

[49] Zhang, H., Balaban, M. ., and Principe, J. C. (2003). Improving pattern recognition of electronic nose data with time-delay neural networks. Sensors and Actuators B: Chemical, 96(1-2), 385-389.

[50] Yamazaki, A., Ludermir, T. B., and De Souto, M. C. P. (2001). Classification of vintages of wine by artificial nose using time delay neural networks. Electronics Letters, 37(24), 1466-1467.

[51] De Vito, S., Castaldo, A., Loffredo, F., Massera, E., Polichetti, T., Nasti, I., ... and Di Francia, G. (2007). Gas concentration estimation in ternary mixtures with room temperature operating sensor array using tapped delay architectures. Sensors and Actuators B: Chemical, 124(2), 309-316.

[52] Duckett, T., Axelsson, M., and Saffiotti, A. (2001). Learning to locate an odour source with a mobile robot. In Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on (Vol. 4, pp. 4017-4022). IEEE.

[53] Fonollosa, J., Sheik, S., Huerta, R., and Marco, S. (2015). Reservoir computing compensates slow response of chemosensor arrays exposed to fast varying gas concentrations in continuous monitoring. Sensors and Actuators B: Chemical, 215, 618-629.

[54] Lukosevicius, M., and Jaeger, H. (2009). Reservoir computing approaches to recurrent neural network training. Computer Science Review, 3(3), 127-149.

[55] Schleif, F. M., Hammer, B., Monroy, J. G., Jimenez, J. G., Blanco-Claraco, J. L., Biehl, M., and Petkov, N. (2016). Odor recognition in robotics applications by discriminative time-series modeling. Pattern Analysis and Applications, 19(1), 207-220.

[56] Persaud, K., and Dodd, G. (1982). Analysis of discrimination mechanisms in the mammalian olfactory system using a model nose. Nature, 299(5881), 352.

[57] Littarru, P. (2007). Environmental odours assessment from waste treatment plants: Dynamic olfactometry in combination with sensorial analysers electronic noses. Waste Management, 27(2), 302-309.

[58] Keller, P. E. (1999, March). Physiologically inspired pattern recognition for electronic noses. In Applications and Science of Computational Intelligence II (Vol. 3722, pp. 144-153). International Society for Optics and Photonics.

[59] Mahmoudi, E. (2009). Electronic nose technology and its applications. Sensors and Transducers, 107(8), 17.

[60] Gardner, J. W., and Bartlett, P. N. (1994). A brief history of electronic noses. Sensors and Actuators B: Chemical, 18(1-3), 210-211.

[61] Kumar, P., Morawska, L., Martani, C., Biskos, G., Neophytou, M., Di Sabatino, S., ... and Britter, R. (2015). The rise of low-cost sensing for managing air pollution in cities. Environment international, 75, 199-205.

[62] Kularatna, N., and Sudantha, B. H. (2008). An environmental air pollution monitoring system based on the IEEE 1451 standard for low cost requirements. IEEE Sensors Journal, 8(4), 415-422.

[63] Galatioto, F., Bell, M., Hodges, N., James, P., and Hill, G. (2011). Integration of low-cost sensors with UTMC for assessing environmental impacts of traffic in urban area. In 18th ITS World CongressTransCoreITS AmericaERTICO-ITS EuropeITS Asia-Pacific.

[64] Galatioto, F., Bell, M. C., and Hill, G. (2014). Understanding the characteristics of the microenvironments in urban street canyons through analysis of pollution measured using a novel pervasive sensor array. Environmental monitoring and assessment, 186(11), 7443-7460.

[65] Arshak, K., Moore, E., Lyons, G. M., Harris, J., and Clifford, S. (2004). A review of gas sensors employed in electronic nose applications. Sensor review, 24(2), 181-198.

[66] Albert, K. J., Lewis, N. S., Schauer, C. L., Sotzing, G. A., Stitzel, S. E., Vaid, T. P., and Walt, D. R. (2000). Cross-reactive chemical sensor arrays. Chemical reviews, 100(7), 2595-2626.

[67] Heeger, A. J. (2001). Semiconducting and metallic polymers: the fourth generation of polymeric materials (Nobel lecture). Angewandte Chemie International Edition, 40(14), 2591-2611.

[68] Pearce, T. C., Schiffman, S. S., Nagle, H. T., and Gardner, J. W. (Eds.). (2006). Handbook of machine olfaction: electronic nose technology. John Wiley and Sons.

[69] Ricco, A. J., Martin, S. J., and Zipperian, T. E. (1985). Surface acoustic wave gas sensor based on film conductivity changes. Sensors and Actuators, 8(4), 319-333.

[70] Schaller, E., Bosset, J. O., and Escher, F. (1998). Electronic noses and their application to food. LWT-Food Science and Technology, 31(4), 305-316.

[71] Grattan, K. T. V., and Sun, T. (2000). Fiber optic sensor technology: an overview. Sensors and Actuators A: Physical, 82(1-3), 40-61.

[72] Tilley, E. (2014). Compendium of sanitation systems and technologies. Eawag.

[73] Barakat, M. A. (2011). New trends in removing heavy metals from industrial wastewater. Arabian Journal of Chemistry, 4(4), 361-377.

[74] Srme, L., and Lagerkvist, R. (2002). Sources of heavy metals in urban wastewater in Stockholm. Science of the Total Environment, 298(1-3), 131-145.

[75] Mateo-Sagasta, J., Raschid-Sally, L., and Thebo, A. (2015). Global wastewater and sludge production, treatment and use. In Wastewater (pp. 15-38). Springer, Dordrecht.

[76] Rice, E. W., Baird, R. B., Eaton, A. D., and Clesceri, L. S. (2012). Standard methods for the examination of water and wastewater. Washington: APHA, AWWA, WPCR, 1496.

[77] Sutton, R. T., Dong, B., and Gregory, J. M. (2007). Land/sea warming ratio in response to climate change: IPCC AR4 model results and comparison with observations. Geophysical Research Letters, 34(2).

[78] Clark, T., Stephenson, T., and Pearce, P. A. (1997). Phosphorus removal by chemical precipitation in a biological aerated filter. Water Research, 31(10), 2557-2563.

[79] Junli, H., Li, W., Nenqi, R., Li, L. X., Fun, S. R., and Guanle, Y. (1997). Disinfection effect of chlorine dioxide on viruses, algae and animal planktons in water. Water Research, 31(3), 455-460.

[80] World Health Organization. (2004). Guidelines for drinking-water quality: recommendations (Vol. 1). World Health Organization.

[81] Arya, F. K., and Zhang, L. (2015). Time series analysis of water quality parameters at Stillaguamish River using order series method. Stochastic environmental research and risk assessment, 29(1), 227-239.

[82] Chuang, M. D., and Yu, G. H. (2007). Order series method for forecasting nonGaussian time series. Journal of Forecasting, 26(4), 239-250.

[83] Hosking, J. R. (1981). Fractional differencing. Biometrika, 68(1), 165-176.

[84] Kurun, A., Yrekli, K., and Cevik, O. (2005). Performance of two stochastic approaches for forecasting water quality and streamflow data from Yeilrmak River, Turkey. Environmental Modelling and Software, 20(9), 1195-1200.

[85] Dawdy, D. R., and Kalinin, G. P. (1970). Mathematical modeling in hydrology.

[86] Ahmad, S., Khan, I. H., and Parida, B. P. (2001). Performance of stochastic approaches for forecasting river water quality. Water research, 35(18), 4261-4266.

[87] Hirsch, R. M., Slack, J. R., and Smith, R. A. (1982). Techniques of trend analysis for monthly water quality data. Water resources research, 18(1), 107-121.

[88] Hipel, K. W., and McLeod, A. I. (1994). Time series modeling of water resources and environmental systems (Vol. 45). Elsevier.

[89] Akaike, H. (1974). A new look at the statistical model identification. IEEE transactions on automatic control, 19(6), 716-723.

[90] Worrall, F., and Burt, T. (2004). Time series analysis of longterm river dissolved organic carbon records. Hydrological Processes, 18(5), 893-911.

[91] Worrall, F., and Burt, T. P. (1999). A univariate model of river water nitrate time series. Journal of Hydrology, 214(1-4), 74-90.

[92] Durdu, . F. (2010). Stochastic approaches for time series forecasting of boron: a case study of Western Turkey. Environmental monitoring and assessment, 169(1-4), 687-701.

[93] Abudu, S., King, J. P., and Bawazir, A. S. (2010). Forecasting monthly streamflow of spring-summer runoff season in Rio Grande headwaters basin using stochastic hybrid modeling approach. Journal of Hydrologic Engineering, 16(4), 384-390.

[94] Hipel, K. W., McLeod, A. I., and Lennox, W. C. (1977). Advances in BoxJenkins modeling: 1. Model construction. Water Resources Research, 13(3), 567-575.

[95] Calabrese, A., Uricchio, V. F., Casale, B., Mauro, R., Blonda, M., PROGETTO MAUI - Monitoraggio continuo per le Acque reflue Urbane ed Industriali per l'eco-industria - Relazione tecnica (2018).

[96] Vergara, A., Fonollosa, J., Mahiques, J., Trincavelli, M., Rulkov, N., and Huerta, R. (2013). On the performance of gas sensor arrays in open sampling systems using Inhibitory Support Vector Machines. Sensors and Actuators B: Chemical, 185, 462-477.

[97] Dua, D. and Karra Taniskidou, E. (2017). UCI Machine Learning Repository [http://archive.ics.uci.edu/ml]. Irvine, CA: University of California, School of Information and Computer Science.

[98] L. Cilenti et al., "Sea water distributed monitoring system: A proposal for architecture and data format," 2018 IEEE International Conference on Environmental Engineering (EE), Milan, 2018, pp. 1-7.

[99] Bonaglia, S., Brchert, V., Callac, N., Vicenzi, A., Fru, E. C., and Nascimento, F. J. (2017). Methane fluxes from coastal sediments are enhanced by macrofauna. Scientific reports, 7(1), 13145.

[100] Bjar Alonso, J., Corts Garca, C. U., and Poch, M. (1993). LINNEO+: a classification methodology for ill-structured domains.

[101] Belanche, L., Snchez, M., Corts, U., and Serra, P. (1992, June). A knowledge-based system for the diagnosis of waste-water treatment plants. In International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems (pp. 324-336). Springer, Berlin, Heidelberg.

[102] Di Lecce, V., Petruzzelli, D., Guaragnella, C., Cardellicchio, A., Dentamaro, G., Quarto, A., ... and Dario, R. Real-time monitoring system for urban wastewater. In Proc. of 2017 IEEE Workshop on Environmental, Energy, and Structural Monitoring Syst.

[103] Boccaletti, Stefano, et al. (2006). Complex networks: Structure and dynamics. Physics reports, 424(4-5), 175-308.

[104] M. Blonda et al., "Innovative Methodology for Detecting of Possible Harmful Compounds for Wastewater Treatment the MAUI Project," 2018 Workshop on Metrology for Industry 4.0 and IoT, Brescia, 2018, pp. 1-6.

[105] Schubert, E., Sander, J., Ester, M., Kriegel, H. P., and Xu, X. (2017). DBSCAN revisited, revisited: why and how you should (still) use DBSCAN. ACM Transactions on Database Systems (TODS), 42(3), 19.

[106] Sander, J., Ester, M., Kriegel, H. P., and Xu, X. (1998). Density-based clustering in spatial databases: The algorithm gdbscan and its applications. Data mining and knowledge discovery, 2(2), 169-194.

[107] Bar-Joseph, Z. (2004). Analyzing time series gene expression data. Bioinformatics, 20(16), 2493-2503.

[108] Scargle, J. D. (1982). Studies in astronomical time series analysis. II-Statistical aspects of spectral analysis of unevenly spaced data. The Astrophysical Journal, 263, 835-853.

[109] Dubes, R., and Jain, A. K. (1980). Clustering methodologies in exploratory data analysis. In Advances in computers (Vol. 19, pp. 113-228). Elsevier.

[110] Hubert, L., and Arabie, P. (1985). Comparing partitions. Journal of classification, 2(1), 193-218.

[111] Rousseeuw, P. J. (1987). Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. Journal of computational and applied mathematics, 20, 53-65.

[112] Estrin, D., Govindan, R., Heidemann, J., and Kumar, S. (1999, August). Next century challenges: Scalable coordination in sensor networks. In Proceedings of the 5th annual

ACM/IEEE international conference on Mobile computing and networking(pp. 263-270). ACM.

[113] Martinez, K., Hart, J. K., and Ong, R. (2004). Environmental sensor networks. Computer, 37(8), 50-56.

[114] Haykin, S. (2005). Cognitive radio: brain-empowered wireless communications. IEEE journal on selected areas in communications, 23(2), 201-220.

[115] Haykin, S., Thomson, D. J., and Reed, J. H. (2009). Spectrum sensing for cognitive radio. Proceedings of the IEEE, 97(5), 849-877.

[116] Chen, G. C., Zhang, L., and Hao, N. M. (2003). Application of neural network PID controller in constant temperature and constant liquid-level system. Micro-computer information, 19(1), 23-24.

[117] Leek, J. T., Scharpf, R. B., Bravo, H. C., Simcha, D., Langmead, B., Johnson, W. E., ... and Irizarry, R. A. (2010). Tackling the widespread and critical impact of batch effects in high-throughput data. Nature Reviews Genetics, 11(10), 733.

[118] Gulshan, V., Peng, L., Coram, M., Stumpe, M. C., Wu, D., Narayanaswamy, A., ... and Kim, R. (2016). Development and validation of a deep learning algorithm for detection of diabetic retinopathy in retinal fundus photographs. Jama, 316(22), 2402-2410.

[119] Baldi, P., and Sadowski, P. J. (2013). Understanding dropout. In Advances in neural information processing systems (pp. 2814-2822).

[120] Rumelhart, D. E., Hinton, G. E., and Williams, R. J. (1985). Learning internal representations by error propagation (No. ICS-8506). California Univ San Diego La Jolla Inst for Cognitive Science.

[121] Kohonen, T. (1990). The self-organizing map. Proceedings of the IEEE, 78(9), 1464-1480.

[122] Jain, A. K., Murty, M. N., and Flynn, P. J. (1999). Data clustering: a review. ACM computing surveys (CSUR), 31(3), 264-323.

[123] Parzen, E. (Ed.). (2012). Time Series Analysis of Irregularly Observed Data: Proceedings of a Symposium Held at Texas A and M University, College Station, Texas February 1013, 1983 (Vol. 25). Springer Science and Business Media.

[124] Maller, R. A., Mller, G., and Szimayer, A. (2008). GARCH modelling in continuous time for irregularly spaced time series data. Bernoulli, 14(2), 519-542.

[125] Taylor, S. J. (2008). Modelling financial time series. world scientific.

[126] Campos, I., Alcaniz, M., Aguado, D., Barat, R., Ferrer, J., Gil, L., ... and Vivancos, J. L. (2012). A voltammetric electronic tongue as tool for water quality monitoring in wastewater treatment plants. Water research, 46(8), 2605-2614.

[127] Winquist, F. (2008). Voltammetric electronic tonguesbasic principles and applications. Microchimica Acta, 163(1-2), 3-10.

[128] Parra, V., Arrieta, . A., Fernndez-Escudero, J. A., Garca, H., Apetrei, C., Rodrguez-Mndez, M. L., and de Saja, J. A. (2006). E-tongue based on a hybrid array of voltammetric sensors based on phthalocyanines, perylene derivatives and conducting polymers: Discrimination capability towards red wines elaborated with different varieties of grapes. Sensors and Actuators B: Chemical, 115(1), 54-61.

[129] Mimendia, A., Gutirrez, J. M., Leija, L., Hernndez, P. R., Favari, L., Muoz, R., and del Valle, M. (2010). A review of the use of the potentiometric electronic tongue in the monitoring of environmental systems. Environmental Modelling and Software, 25(9), 1023-1030.

[130] Nicolas, J., and Romain, A. C. (2004). Establishing the limit of detection and the resolution limits of odorous sources in the environment for an array of metal oxide gas sensors. Sensors and Actuators B: Chemical, 99(2-3), 384-392.

[131] Mumyakmaz, B., zmen, A., Ebeolu, M. A., Taaltn, C., and Grol, . (2010). A study on the development of a compensation method for humidity effect in QCM sensor responses. Sensors and Actuators B: Chemical, 147(1), 277-282.

[132] Ziyatdinov, A., Marco, S., Chaudry, A., Persaud, K., Caminal, P., and Perera, A. (2010). Drift compensation of gas sensor array data by common principal component analysis. Sensors and Actuators B: Chemical, 146(2), 460-465.

[133] Chong, C. Y., and Kumar, S. P. (2003). Sensor networks: evolution, opportunities, and challenges. Proceedings of the IEEE, 91(8), 1247-1256.

[134] Krantz-Rlcker, C., Stenberg, M., Winquist, F., and Lundstrm, I. (2001). Electronic tongues for environmental monitoring based on sensor arrays and pattern recognition: a review. Analytica chimica acta, 426(2), 217-226.

[135] Madou, M. J., and Morrison, S. R. (2012). Chemical sensing with solid state devices. Elsevier.

[136] Kim, J., Lim, J. S., Friedman, J., Lee, U., Vieira, L., Rosso, D., ... and Srivastava, M. B. (2009, June). Sewersnort: A drifting sensor for in-situ sewer gas monitoring. In Sensor, Mesh and Ad Hoc Communications and Networks, 2009. SECON'09. 6th Annual IEEE Communications Society Conference on (pp. 1-9). IEEE.

[137] Olsson, J., Ivarsson, P., and Winquist, F. (2008). Determination of detergents in washing machine wastewater with a voltammetric electronic tongue. Talanta, 76(1), 91-95.

[138] Gallardo, J., Alegret, S., and del Valle, M. (2004). A flow-injection electronic tongue based on potentiometric sensors for the determination of nitrate in the presence of chloride. Sensors and Actuators B: Chemical, 101(1-2), 72-80.

[139] Gardner, J. W., Shin, H. W., Hines, E. L., and Dow, C. S. (2000). An electronic nose system for monitoring the quality of potable water. Sensors and Actuators B: Chemical, 69(3), 336-341.

[140] Canhoto, O. F., and Magan, N. (2003). Potential for detection of microorganisms and heavy metals in potable water using electronic nose technology. Biosensors and Bioelectronics, 18(5-6), 751-754.

[141] Dewettinck, T., Van Hege, K., and Verstraete, W. (2001). The electronic nose as a rapid sensor for volatile compounds in treated domestic wastewater. Water Research, 35(10), 2475-2483.

[142] Lamagna, A., Reich, S., Rodrguez, D., Boselli, A., and Cicerone, D. (2008). The use of an electronic nose to characterize emissions from a highly polluted river. Sensors and Actuators B: Chemical, 131(1), 121-124.

[143] Ester, M., Kriegel, H. P., Sander, J., and Xu, X. (1996, August). A density-based algorithm for discovering clusters in large spatial databases with noise. In Kdd (Vol. 96, No. 34, pp. 226-231).

[144] Lloyd, S. (1982). Least squares quantization in PCM. IEEE transactions on information theory, 28(2), 129-137.

[145] LeCun, Y. A., Bottou, L., Orr, G. B., and Mller, K. R. (2012). Efficient backprop. In Neural networks: Tricks of the trade (pp. 9-48). Springer, Berlin, Heidelberg.

[146] Breiman, L. (2001). Random forests. Machine learning, 45(1), 5-32.

[147] Safavian, S. R., and Landgrebe, D. (1991). A survey of decision tree classifier methodology. IEEE transactions on systems, man, and cybernetics, 21(3), 660-674.

[148] Raileanu, L. E., and Stoffel, K. (2004). Theoretical comparison between the gini index and information gain criteria. Annals of Mathematics and Artificial Intelligence, 41(1), 77-93.

[149] Strobl, C., Boulesteix, A. L., Zeileis, A., and Hothorn, T. (2007). Bias in random forest variable importance measures: Illustrations, sources and a solution. BMC bioinformatics, 8(1), 25.

[150] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., ... and Vanderplas, J. (2011). Scikit-learn: Machine learning in Python. Journal of machine learning research, 12(Oct), 2825-2830.

[151] CUTLER, B. D. W., and CRUMP, L. M. (1929). Carbon dioxide production in sands and soils in the presence and absence of amoebae. Annals of Applied Biology, 16(3), 472-482.

[152] Bullmore, E., and Sporns, O. (2009). Complex brain networks: graph theoretical analysis of structural and functional systems. Nature Reviews Neuroscience, 10(3), 186.

[153] Cardellicchio, A., Lombardi, A., and Guaragnella, C. (2018). An Iterative Complex Network Approach for Chemical Gas Sensor Array Characterization. Submitted to The IET Journal of Engineering.

[154] Buitinck, L., Louppe, G., Blondel, M., Pedregosa, F., Mueller, A., Grisel, O., ... and Layton, R. (2013). API design for machine learning software: experiences from the scikit-learn project. arXiv preprint arXiv:1309.0238.

[155] Goodfellow, I., Bengio, Y., Courville, A., and Bengio, Y. (2016). Deep learning (Vol. 1). Cambridge: MIT press.

[156] Kingma, D. P., and Ba, J. (2014). Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980.

[157] Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. (2014). Dropout: a simple way to prevent neural networks from overfitting. The Journal of Machine Learning Research, 15(1), 1929-1958.

[158] Guyon, I., Weston, J., Barnhill, S., and Vapnik, V. (2002). Gene selection for cancer classification using support vector machines. Machine learning, 46(1-3), 389-422.

[159] Ren, V., Cardellicchio, A., Politi, T., Guaragnella, C., and D'Orazio, T. (2016, February). Exploiting ambiguities in the analysis of cumulative matching curves for person re-identification. In Proceedings of the 5th International Conference on Pattern Recognition Applications and Methods (pp. 484-494). SCITEPRESS-Science and Technology Publications, Lda.

[160] Ren, V., Cardellicchio, A., Politi, T., Guaragnella, C., and DOrazio, T. (2016, February). Comparative Analysis of PRID Algorithms Based on Results Ambiguity Evaluation. In International Conference on Pattern Recognition Applications and Methods (pp. 230-242). Springer, Cham.

[161] Dentamaro, G., Cardellicchio, A., and Guaragnella, C. (2016, December). Real time Artificial Auditory Systems for cluttered environments. In Pattern Recognition (ICPR), 2016 23rd International Conference on (pp. 2234-2239). IEEE.

[162] Quarto, A., Soldo, D., Gemmano, S., Dario, R., Di Lecce, V., Guaragnella, C., ... and Lombardi, A. IoT and CPS applications based on wearable devices. A case study: monitoring of elderly and infirm patients.

[163] Cardellicchio, Angelo, Rita Dario, Vincenzo Di Lecce, Cataldo Guaragnella, Angela Lombardi, Lucia Mongelli, Alessandro Quarto, and Domenico Soldo. "Evaluation of smartphone usage in neurological pathologies diagnosis."

[164] Lombardi, A., Tangaro, S., Bellotti, R., Cardellicchio, A., and Guaragnella, C. (2017, September). Identification of Die Hard Nodes in Complex Networks: A Resilience Approach. In Italian Workshop on Artificial Life and Evolutionary Computation (pp. 257-268). Springer, Cham.