POLITECNICO DI TORINO

Master's Degree in ICT for Smart Societies

Master's Degree Thesis

Smart Manufacturing and Predictive Maintenance driven by Machine Learning

Supervisors

Prof. Danilo GIORDANO

Prof. Marco MELLIA

Prof. Elena Maria BARALIS

Candidate

Luca TOMAINO

23rd July 2020

Acknowledgements

I definitely want to say "thank you" to Professor Giordano and Professor Mellia, who followed this thesis, because not only did they guide and advise me in the right way, but they also made me feel an integral part of the project. Special thanks also to Mr. Genchi and the entire working group at "Centro Ricerche Fiat" for the valuable advice and the absolute availability that they have had towards me. This project, for me, represents not only the end of my university career, but also the end of a long and intense journey, during which I learned a lot and, above all, I changed a lot. For this reason, it is my duty to thank my family and my closest friends for the support and help they have given me, especially in the most difficult moments, and along with them, a thank you also goes to those who have been part of this journey for a more or less long time.

Summary

This project focuses on the development of a data-driven methodology that aims to gain more knowledge about welding processes performed on car bodies. This methodology is fundamental to the transition from a periodic maintenance scheme to a predictive maintenance one, which would allow the optimisation of the maintenance scheme itself and the increase of money and energy savings and of the quality of the production. Moreover, the methodology draws inspiration from Data Mining concept, in order to better match with the technologies and techniques that are being developed in the context of industry 4.0.
In the project, the performance of unsupervised analysis with clustering algorithms and of aggregation and supervised analysis is done in order to gain a deeper knowledge of the provided data and in order to understand if there is the possibility to introduce predictive maintenance schemes to replace the existing periodic ones. Moreover, there is also the analysis of how the presence of an anomaly in the production process influences the following operations in terms of production quality and energy consumption.
The developed data-driven methodology is inspired to the so-called "knowledge discovery in databases" process that is typical of data mining analysis. This methodology consists of a pipeline of operations that includes all the steps contained in the just mentioned process, but also adds some tasks necessary for the data to be analysed. Moreover, the methodology includes a clustering part in which a customised algorithm (based on the k-means one) is developed in order to better deal with time-series clustering, that is characterised by the presence of time-series instead of single data points.
The validation of a customised algorithm, the comparison between that and a standard one and the analysis of anomalies are then the milestones of this project and they all contribute to an exploratory analysis of the provided data in order to understand if there is the possibility of the transition from periodic to predictive maintenance that has been previously introduced.

Table of Contents

List of Tables

List of Figures

Acronyms

AI
Artificial Intelligence

ARMA
Auto-Regressive Moving Average

GUI
Graphical User Interface

HMM
Hidden Markov Model

IoT
Internet of Things

KDD
Knowledge Discovery from Data

ML
Machine Learning

MVST
Multivariate Time Series

PdM
Predictive Maintenance

PLC
Programmable Logic Controller

SCADA

Supervisory Control and Data Acquisition

SMEs

Small- to Medium- scale Enterprises

SSE

Sum of Squared Errors

TDC

Tip Dress Counter

Chapter 1

Introduction

1.1 Background of the project

Nowadays, the world is is in the midst of a new industrial revolution and the concept of "Industry 4.0" is extremely widespread. This term was actually coined in Germany in 2011 and refers to the application of new technologies and methodologies to the industrial world, with the dual aim of improving the quality of work and production and increasing productivity, also creating new business models. Among the new technologies and methodologies, it is necessary to include some Data Mining techniques, the Internet of Things, Cloud solutions, but also the concepts of Predictive Maintenance and Smart Manufacturing (that together with the concept of Smart Factory incorporates what has just been listed).

Focusing more on the project presented in this work, it is important to clarify the concept of "Predictive Maintenance": according to [1], "predictive maintenance ("PdM") techniques are designed to help determine the condition of in-service equipment in order to estimate when maintenance should be performed". In this way, it is possible to achieve relevant savings in terms of costs, with respect to common maintenance processes based on routines or time. In particular, predictive maintenance is performed only when needed and therefore the condition of what has to be maintained should be periodically monitored.

This thesis, in particular, focuses on the analysis of data coming from welding processes performed on car bodies in a specific production plant. The work is referred to the operations performed at the Mirafiori Plant in Torino on the car bodies of the Maserati "Levante" model. Each welding process is performed by a robot on whose tip there is an electrode that is consumed with the advancement of the welds and that therefore periodically needs maintenance, which consists of two

operations:

- electrode dressing to remove accumulated dirt;
- electrode replacement after that some dressings have been performed.

Currently, the maintenance operations are performed on a regular basis, always with the same frequency (i.e. after a certain number of welds there is a dressing and after a certain number of dressings there is a replacement) and each robot features different periods for the operations themselves.

1.2 Goal of the project

This project is about the development of a data-driven methodology that aims to gain more knowledge about welding processes performed on car chassis. This methodology has a key role in helping the transition towards a predictive maintenance model through which it could be possible to:

- optimise the maintenance of the machinery and then improve the quality of the production;
- increase money savings and efficiency, for instance by reducing the frequency of tools replacements, then decreasing the energy consumption.

The just mentioned methodology fits into the context of "Data Mining", which is better described in chapter 2, even a first definition can be already provided. According to [2], "Data Mining is the process of discovering patterns in large data sets involving methods at the intersection of machine learning, statistics, and database systems." Data Mining is bonded to the concept of Data Science, which means "Extracting meaning from very large quantities of data" according to the definition given by D.J. Patil. In particular, it consists of extracting information that is implicit, not known in advance and that could be useful; extraction is performed automatically through proper algorithms and what is extracted from data is organised in patterns.

Among the various Data Mining techniques, the presented work is mainly related to clustering and more specifically to time-series clustering. As will be better explained in section 2.2, clustering can be described as an unsupervised data mining technique that consists of forming homogeneous groups (called "clusters") of objects. Groups are characterised by a minimum inter-group similarity and a maximum intra-group one. Time series clustering is the application of clustering algorithms to time-series objects and features some issues and peculiarities that must be solved to get satisfying results. Therefore, the main difference with respect to standard clustering is that the objects are not single points described by coordinates, but curves that develop in time (that's why they are addressed as "time-series").

1.3 Related work

Predictive maintenance and smart manufacturing

Krupitzer et al, in [3], put on evidence two main issues related to maintenance and more specifically yo the way it is currently executed in many industrial contexts:

- maintenance operations are responsible for a relevant percentage of industrial production costs (15 % to 70 %);
- nowadays, in the world, most of the industrial situations are characterised by outdated maintenance schemes such as "run-to-failure" or statistical-driven ones.

The proposed solution to those problems of the production is then PdM. The authors say that "PdM is based on the idea that certain characteristics of machinery can be monitored and the gathered data can be used to derive an estimation about the remaining useful life of the equipment". This is very important as it is possible to, apart from increasing money and energy savings, improve production quality in two ways:

- reduction of the overall maintenance processes by avoiding too early and/or unnecessary maintenance on tools whose condition is still good (even if they are supposed to go through maintenance according to periodic schedules);
- avoidance of too late maintenance on tools whose degradation occurs faster than what periodic maintenance schedules have predicted.

PdM has become increasingly widespread in recent years and is responsible for the advent of the so-called "industry 4.0". This expression is used to refer to the fourth revolution of manufacturing processes, concerning the application of machine learning (ML) techniques to industrial contexts with the aim of improving productivity and product quality. There are many interesting examples of the application of PdM techniques to real cases and contexts and this proves the effectiveness of those techniques, since in most of the situations beneficial effects have been identified.

In the work of Kiangala et al [4], PdM is applied to a small-scale bottling plant in order to demonstrate that those techniques are not only suitable for big plants, but also for Small- to Medium- scale Enterprises (SMEs). In this example, it is possible to detect early faults regarding a conveyor motor and to generate a proper and effective PdM schedule that depends on the detection itself. The authors analyse the usage of a programmable logic controller (PLC) together with a series of sensors, an supervisory control and data acquisition (SCADA) system and a graphical user interface (GUI). This set of tools is the key for the development of

the just mentioned detection-based PdM scheme.

Another interesting solution is that suggested by Traini et al [5], who exploit the combination of ML and Internet of Things (IoT) to create a PdM framework to be applied on a milling cutting-tool, also including a wear monitoring system. In particular, this work is about the usage of some types of regression (e.g. Linear, Neural Network etc.) to allow the creation of a PdM framework in order to:

- provide an exact forecast of when machinery will need maintenance;
- optimise the production process;
- improve human-machine interaction.

Moreover, the presented environment can be also used in different contexts, apart from the one described in the paper.

The paper presented by Leong et al [6] is focused on a similar topic with respect to that of Traini et al, since it proposes an IoT-based PdM solution to be applied on Smart Manufacturing systems. The difference, in this case, is that the application is performed on a welding machine (then this case is also correlated to the one presented in this thesis) instead of a milling cutting-tool; in particular, data are collected using a bunch of smart sensors mounted on the machine itself and then the monitoring is performed with statistical process control methods. After that, the application of ML techniques reveals "hidden correlations in the data sets and detects abnormal data patterns" that are then used to identify (via classification approaches) the type of manufacturing processes, that can be regular or can present a failure (whose most important contributing variable is identified, too).

A final example of PdM application is the one provided by Hwang et al [7], who use ML techniques like Support Vector Machines (SVM) and Restricted Boltzmann Machine (RBM) to distinguish between processes characterised by a fault and the regular ones, thus creating another PdM scheme.

Chapter 2

Introduction to Data Mining and methodology explanation

2.1 Basic notions

As reported in [8], Data Mining is part of a larger process called Knowledge Discovery from Data (KDD). This process is composed by the following steps, that are also described in figure 2.1:

- selection of a part of interest of the data and preprocessing of the selection itself; preprocessing consists of
  - data cleaning, to reduce noise influence, solve inconsistencies and remove outliers;
  - data integration, to integrate data coming from different sources with themselves and with metadata, also identifying data conflicts;
- transformation of the preprocessed data; it is a two-steps process:
  - first of all a mapping of the data is done, to assign elements from source base to destination in order to capture transformations;
  - then, the actual transformation program is generated;
- data mining, during which relevant data are turned into patterns and then the purpose of the model is set by choosing between classification and characterisation;

- interpretation of the extracted pattern to get some knowledge about obtained information.

There are two kind of analysis techniques: descriptive methods (exact models that describe data) and predictive ones (that instead use known variables to predict future ones). In particular, there are supervised approaches (in which the labels of the pattern to create are known) such as classification (whose goal is precisely the prediction of a class label) and unsupervised ones, such as clustering, that is the one used in this work and will be described in the next section.

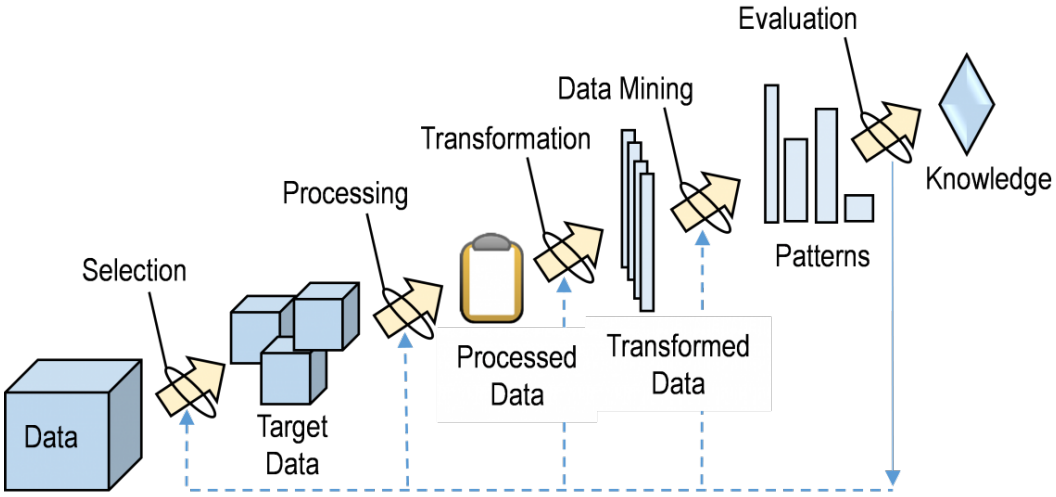


Figure 2.1: KDD process description

2.2 Clustering

As written in [9], cluster analysis can be described as searching groups of objects such that the objects that are contained in the same group features high correlation among themselves and a clear difference with respect to the objects present in the other groups. According to [10], clustering can also be addressed as "the task of grouping a set of objects in such a way that objects in the same group (called a cluster) are more similar (in some sense) to each other than to those in other groups (clusters)". In particular, the similarity between two objects is computed by evaluating their distance. Therefore, the shorter the distance, the higher the correlation between two objects. The distance itself can be evaluated in different ways as there are various types of distance functions, such as:

- Euclidean distance, computed as $d(x, y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$

- Manhattan distance, computed as $d(x, y) = \sum_{i=1}^{n} |(x_i - y_i)|$
- various correlation-based distance (Pearson, Eisen cosine, Spearman...)

2.2.1 Why clustering?

Even if there would be many alternatives to choose as main technique for this work, clustering seems to be the best one. In fact, in this work the main aspect is the absence of a label for the given data and then an exploratory analysis of the data themselves is requested. Clustering fits perfectly with this need for data exploration since it does not need the presence of labels and that is why it has been chosen.

2.2.2 Clustering typologies

The variety of situations in which clustering can be implied is huge and there is also a large amount of clustering techniques. Looking again at [9], the main distinction can be made between:

- partitional clustering, that consists in "a division of data objects into non-overlapping subsets (clusters) such that each data object is in exactly one cluster";
- hierarchical clustering, through which "a set of nested clusters is organized as a hierarchical tree".

Sets of clusters can also be exclusive or non-exclusive (in the second case, points may belong to different clusters at the same time), fuzzy or non-fuzzy (if clustering if fuzzy, each point has a coefficient whose value is between 0 and 1 and that weighs the probability of that point to belong to any cluster), partial or complete (if partial, only a part of the data is considered for clustering) and heterogeneous or homogeneous, but these are secondary distinctions that can be applied. A classification is also possible for what concerns types of clusters; in fact, there are some typologies:

- well-separated clusters (any point is closer to all those in the same cluster than to those belonging to any other cluster);
- center-based clusters (any object in a cluster is closer to the cluster's "center": it most representative point);
- contiguous clusters (a point in a cluster is closer to one or some others in the same cluster than to any point outside the cluster itself);

- density-based clusters (a cluster is a dense region of points and different dense regions of points are separated by non-dense ones);
- property/conceptual clusters (clusters are created looking at particular properties or at points expressing the same concept);
- clusters described by an objective function.

2.2.3 In this project: custom-distance k-means clustering

In the proposed work, the main technique that has been implied is a modified version of the k-means algorithm. This algorithm creates center-based clusters, which are (as described in section 2.2.2) "sets of objects such that an object in a cluster is closer (more similar) to the “center” of a cluster, than to the center of any other cluster" [9]. The just mentioned center of a cluster can be a centroid (therefore the average of all the points in the cluster) or a medoid (which can be addressed as the most representative point of a cluster). Algorithm 1 shows a basic implementation of k-means. It is important to put on evidence some aspects:

- the process goes on for a certain number of iteration until convergence;
- centroids are usually initialised randomly, even if the initialisation itself is quite important for the convergence (in fact, performing the initialisation in a non-random way means increasing the possibility to have a smaller number of iteration and better results);
- similarity among objects of a cluster is measured through a distance metric that can be one of those presented earlier in this section.

Algorithm 1: Basic k-means algorithm

```
Select K points as the initial centroids;
while Centroids do not change do
    Form K clusers by assigning all points to the closest centroid;
    Recompute the centroid of each cluster;
end
```

As said before, the proposed work presents a modified version of the k-means algorithm which better suits the implication of time series instead of single data points. The modification regards the distance metric to evaluate the similarity among objects that leads to their assignment to a certain cluster. In a standard k-means algorithm, as written in algorithm 1, the Euclidean distance is the one

preferred.

In this particular case, instead, the distance should be evaluated between current and voltage (but also resistance, if needed) curves, then using a simple Euclidean distance is not the ideal solution, since the assignment to a cluster should be performed considering all the provided data for a welding process. To fix this, the idea is to combine the different curves of each file after the computation of the distance among signals. The distance metric is then the following:

$$d(\mathbf{k}, \mathbf{j}) = d(v_k, v_j) + d(i_k, i_j) \tag{2.1}$$

where **k** and **j** are two signals and **v** and **i** are their voltage and current components. Moreover, the distance calculated between the same component of two signals is defined as

$$d(x_k, x_j) = \frac{1}{N} \sum_{i=1}^{N} d(i) \tag{2.2}$$

with the distance d(i) addressed as the distance between two values at the same time instant for a component:

$$d(i) = \sqrt[m]{(x_k(i) - x_j(i))^m} \tag{2.3}$$

2.2.4 Clustering evaluation

The evaluation of a clustering process, and in particular of the k-means one, can be done with some techniques, even if two of them are the most popular ones.

The first technique is the evaluation of the sum of squared errors (SSE). The error is defined as the distance of each point from the nearest cluster and the SSE is computed as:

$$SSE = \sum_{i=1}^{K} \sum_{i \in C_i} d^2(m_i, x) \tag{2.4}$$

in which K is the number of clusters, C_i is the current cluster, x is a data point in cluster C_i itself and m_i is its most representative point. It is clear that the goal is to have the lowest SSE possible, without having to increase the number of clusters k too much. In fact, the ideal situation is that of a low number of clusters and at the same time a small SSE.

The second important technique to evaluate how good is the performance of a clustering algorithm is the one related to the silhouette index, which is particularly useful to evaluate how consistent clusters are within themselves. For each datapoint i, two values are calculated:

- a(i), the average distance between the considered object and any other one belonging to its same cluster;
- b(i), the smallest average distance from i to all the points belonging to any other cluster, apart from the one to which i actually belongs.

The two values are computed as:

$$a(i) = \frac{1}{|C_i| - 1} \sum_{j \in C_i, j \neq i} d(i, j) \tag{2.5}$$

$$b(i) = \min_{k \neq i} \frac{1}{|C_k|} \sum_{j \in C_k} d(i, j) \tag{2.6}$$

with C_i cluster to which point i belongs, C_k any cluster to which i does not belong and $d(i, j)$ distance metric used in the algorithm (in this case, the custom one presented in equation 2.1). After that $a(i)$ and $b(i)$ have been computed, the silhouette index for the data point i under analysis can be evaluated as

$$s(i) = \frac{b(i) - a(i)}{\max a(i), b(i)} \qquad if |C_i| > 1 \tag{2.7}$$

or as

$$s(i) = 0 \qquad if |C_i| = 1 \tag{2.8}$$

For each k number of clusters, the silhouette index $s(i)$ is computed for all the data points, then the average silhouette score is calculated as the mean value of silhouette over all the data points, for the specific k. In the end the value of k that features the highest average silhouette score is the best one for that particular dataset.

Chapter 3

Data Overview and Data Analysis

3.1 Presentation of the system

In section 1.1 it has been introduced that this work focuses on the data coming from welding operation performed by proper robots at FCA's Mirafiori plant in Torino, in which the Maserati "Levante" model is produced. Apart from the specification of the model or the position of the plant, what is important to describe is the composition of the system. As said, in the plant there are some robots that perform welding operations on car bodies. Bodies are mapped on a set of spots distinguished by name and to each robot a determined subset of spots is assigned. Moreover, each robot is characterised by the presence of one or more clamps that actually perform the welds through electrodes positioned on their tips.

Each robot has a welding measurement tool able to evaluate and collect various information related to the welding processes. This is actually the part of the system that allows data collection and then all the other tasks described in the project. In particular, collected data are gathered in proper JSON files called "WeldLogs".

The data collected for this project are referred to the period between November 2019 and February 2020 and the focus has been on three of the robots present in the plant itself.

3.2 Overview of the data

Provided JSON files contain various fields regarding actual data and some metadata related to the welding process under study. The most significant ones are:

- the robot ID;
- the spot ID;
- the welding process timestamp, indicating date, hour, minutes, seconds and microseconds;
- the "tip dress counter" (TDC), which how many bodies have been welded since the last electrode replacement (an example is provided in figure 3.1);
- the current nominal value (i.e. the current value that the welding process should reach at regime);
- the voltage and the current curve of the weld, presented as series of values over time;
- a field called "wear", expressing the absolute consumption of the electrode (this is not a very reliable parameter, according to the domain experts, though);
- the (possible) time instant at which an expulsion of material occurs (the expulsion of material can be due to different reasons and it is considered a major anomaly of the process);
- a progressive number to count the spot's name, whose field in the JSON files is called "progNo".

Together with the "WeldLog" file, if an expulsion occurs during the process, a related "FaultLog" JSON file is generated. It contains a part of the information of the "WeldLog" file itself that are more significant to characterise the anomaly.
It is also important to underline that each welding process, depending on the spot and on the robot that executes it, can last between 200 to 800 milliseconds and that the sampling frequency is 1 kHz.

3.3 Materials and welding techniques

For each robot and clamp information about dressing and electrode replacement frequency are provided, together with a complete list of points, each of them defined by thickness and specification of the material. It is possible to make a distinction

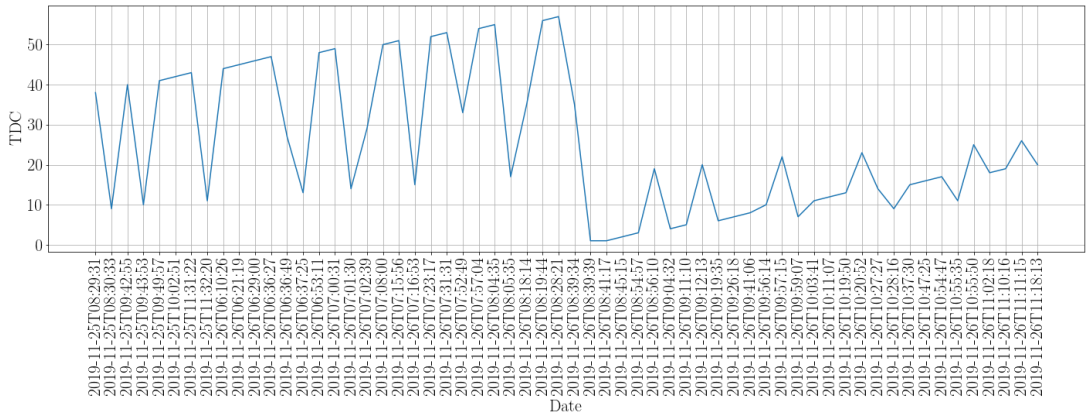


Figure 3.1: Example of the trend of TDC.

based on the number of sheets that compose a spot and this aspect also influences the type of welding technique used for that particular spot. Basically, a spot can be composed by two or three sheets of different materials and thicknesses and can also imply more malleable materials or high-resistance steel (which is usually the one causing more problems in terms of welding faults). According to the spot position and the type of materials, there are different welding techniques and it is now time to analyse two of them.

- The first one (described in figure 3.2) is a three-step technique, composed by:
  - a warm-up phase during which the maximum current/voltage value is reached;
  - a stopping phase in which the electrodes are put in a stopping state for some milliseconds (voltage and current down to 0);
  - a steady-state phase in which the nominal current/voltage value can be reached since the electrode is now warmed-up and ready to perform the weld.
- The second technique (shown in figure 3.3) is characterised, instead, by a warm-up phase during which there is an increase from 0 to the nominal current/voltage value and by a steady-state phase consisting of the proper welding, maintaining the values of current and voltage as constant as possible.

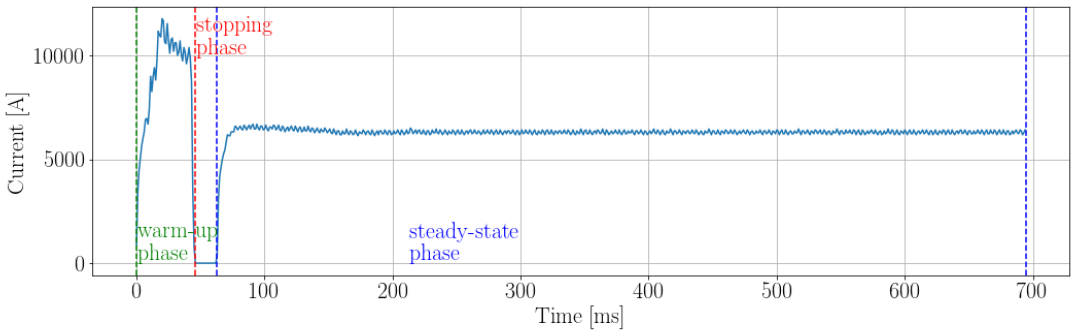


Figure 3.2: Example of the first welding technique.

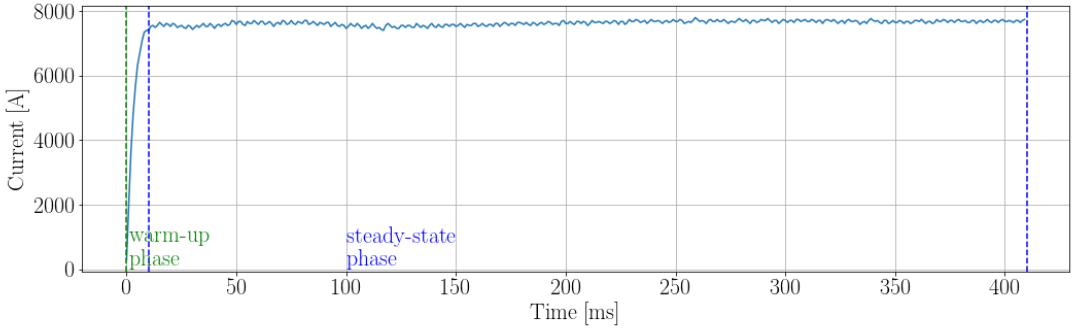


Figure 3.3: Example of the second welding technique.

3.4 Data used in this work and related experiments

The big amount of available data allows the execution of several experiments in order to explore the data themselves in a complete and exhaustive way. In particular, various sets of that are considered and they will be described in the following sections.

3.4.1 Adjustment weldings

The first experiment led in the project is related to a particular set of spots whose related JSON files are all characterised by the same progressive number. These points are quite interesting as they represent a no-load weld in which the the two parts of the welding tip are simply brought closer. This is usually done after a dressing in order to check that everything is still working and those welding process

are characterised by a very short duration of about 100 ms (while, as already said in section 3.2, that of a normal process usually lasts between 200 and 800 milliseconds). The data related to adjustment weldings will be then used with the custom-distance clustering algorithm, especially to understand if the TDC influences somehow the generated voltage and current curves.

3.4.2 Welding processes after a dressing

The first experiment actually regards a particular situation in which there are not proper welding processes. The second test, then, actually deals with more concrete data and is run on the files related to a very significant spot, that is the one on which one of the robots works after every dressing, therefore starting a new sequence of welding points. Working on that, then, it is possible to understand if the increase of the TDC or if the distance from the last dressing (in terms of welded bodies) can influence somehow the performance of the welding processes and the quality of the production. This will be verified through clustering, too, but above all by comparing the clustering results with the actual TDC and distance-from-dressing values, aiming to find relevant patterns in the data.

3.4.3 Supervised analysis

The third experiment is a bit different from the other ones executed as it implies a supervised analysis instead of the already described unsupervised one, related to clustering. In fact, this time what is done is to form groups by classifying them (i.e. dividing files according to the labels assigned to them by looking at TDC and distance form dressing) and then correlating the generated subsets of data with the already mentioned metadata to understand if there is a pattern or a particular behaviour that can be detected.

3.4.4 In-sequence analysis

The fourth experiment regards the analysis of a complete sequence of welding processes done by another robot of those present in the plant. This is an analysis that has been requested by the technicians of the company with which this project is developed and has the goal of understanding if it is possible to detect strange behaviours inside a complete sequence, instead of evaluating processes that belongs to the same welding spot and that then are linked to different welding sequences.

3.4.5 Fault analysis

The first experiments have the final goal of executing a certain clustering algorithm, proving its feasibility and comparing its results with some relevant metadata. The final aim is then to try and understand if it is possible to determine some predictive maintenance schemes to replace the current periodic ones. The fourth and fifth experiment, instead, are related to the analysis of faults, since, as written in section 3.2, one of the fields of the JSON file provided for each welding processes contains the possible time instant at which a failure (consisting in an expulsion of material from the welding place). In particular, the analysis of those anomalies will be correlated with the study of the energy consumption of regular and non regular processes, to investigate if the presence of a fault has an influence on the quality of the process. The analysis regarding the energies will be then brought one step further by choosing one point characterised by a medium percentage of anomalies and trying to understand if the welding process right before and right after a fault for that point itself present some particular behaviour.

Chapter 4

Working pipeline

Taking inspiration from the KDD process described in section 2.1 a pipeline describing the flow of the operation done for each experiment is built.

4.1 Data visualisation

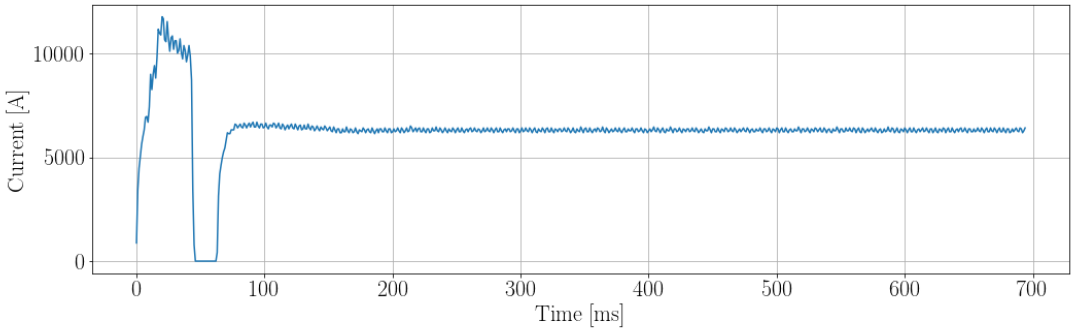
The first thing to do is to visualise the data on which the experiment is going to be performed. In particular, as seen before in section 3.2, each curve is first of all visualised as it is, therefore without any kind of filter applied on it. In this way it is possible to evaluate the welding technique, the average length of the welding processes and some other important characteristics. Figure 4.1 shows two examples (one per welding technique) of "raw" data visualisation.

4.2 Data preprocessing

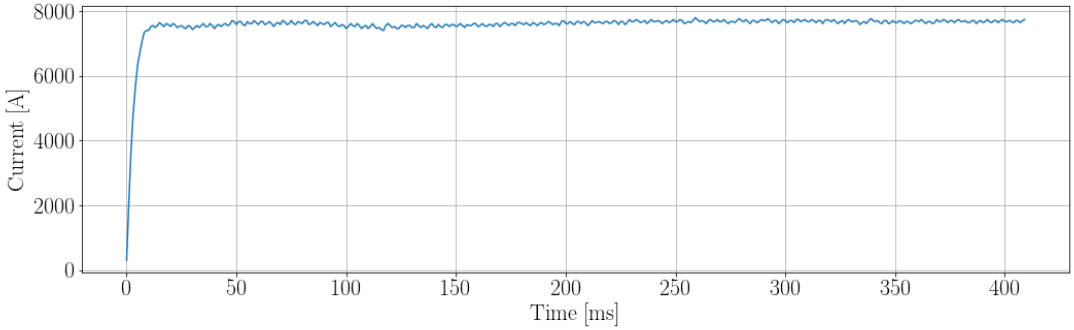
Usually, there is no missing information in provided data, even if they need some preprocessing operations to be done before going on with the analysis.

4.2.1 Frequency analysis

Looking at figure 4.2, it is clear that each current curve (but this is valid also for voltage ones) is characterised by noise throughout the whole welding time. It is then necessary to filter out this noise, but to do that it is needed to understand how to make the curves smoother. The solution is to perform a frequency analysis in order to determine if it is possible to apply a low-pass filter that could cut out the already mentioned noise, providing cleaner curves that are easier to visualise and to analyse. As shown in figure 4.3, the provided example of current (in blue)



((a)) Three-steps welding technique.



((b)) Two-steps welding technique.

Figure 4.1: Examples of data visualisation.

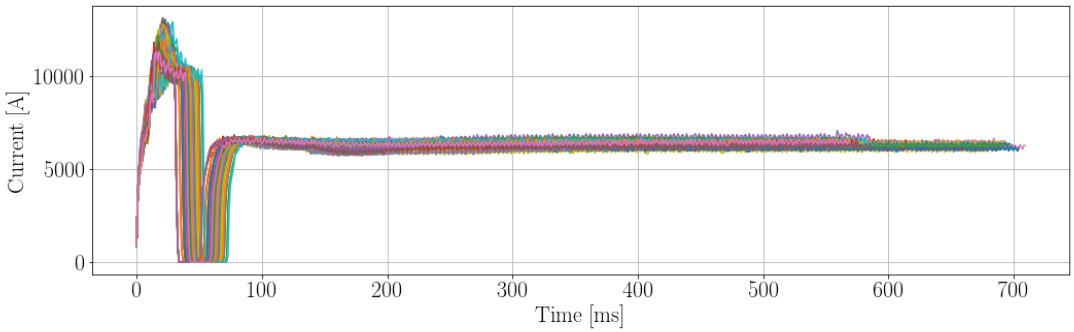


Figure 4.2: Example of noisy current curves to be smoothed.

and voltage (in red) power spectrum show some minor lobes that can be actually filtered out in order to make the curves smoother.

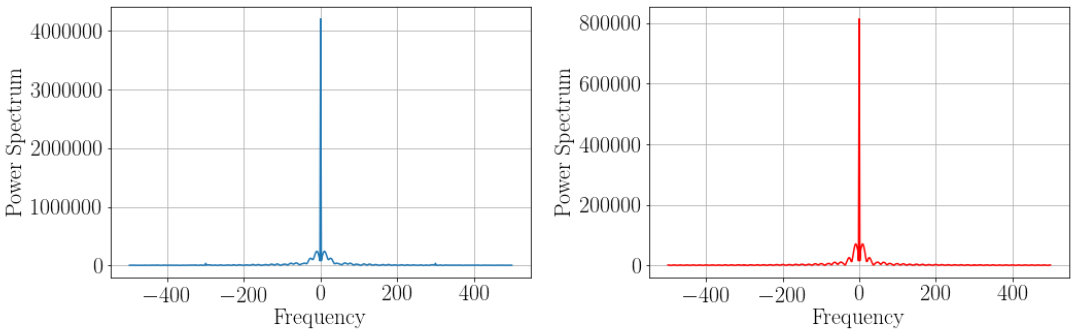


Figure 4.3: Power spectrum example for current (in blue) and voltage (in red).

4.2.2 Smoothing

As already said looking at figure 4.3, it is possible to apply a low-pass filter with the aim of making the curves smoother. In particular, the idea is to use a filter that implies the exponential moving average function, defined as it follows:

$$\bar{X}_{i+1} = \alpha * y_i + (1 - \alpha) * \bar{X}_i \tag{4.1}$$

where $\bar{X}_{i+1}$ is the value of the exponential moving average at time step $i + 1$, EMA_i is the same thing, but at time step i, α is the coefficient that determines how relevant is the smoothing on the curve and y_i is the current value at time i. It is important to underline that coefficient α is set to 0.75 in this project, since higher values would cause information loss, but lower ones would make the filter almost useless as the noise would not be cut out. In figure 4.4 it is possible to appreciate the effect of the filter on a curve, which is visualised before and after the application of the filter itself. The reduction of the noise, in particular, is quite useful for two main reasons:

- the possibility to perform a more precise clustering, since there are not too many spikes and values oscillating up and down and then each object can be assigned to a cluster more easily;
- the possibility to perform a better windowing of the curves, as it will be better described in section 4.5.

4.3 Features extraction

The extraction of the features to be used when executing a clustering algorithm is a crucial process, since the right set of features can provide very good results, but

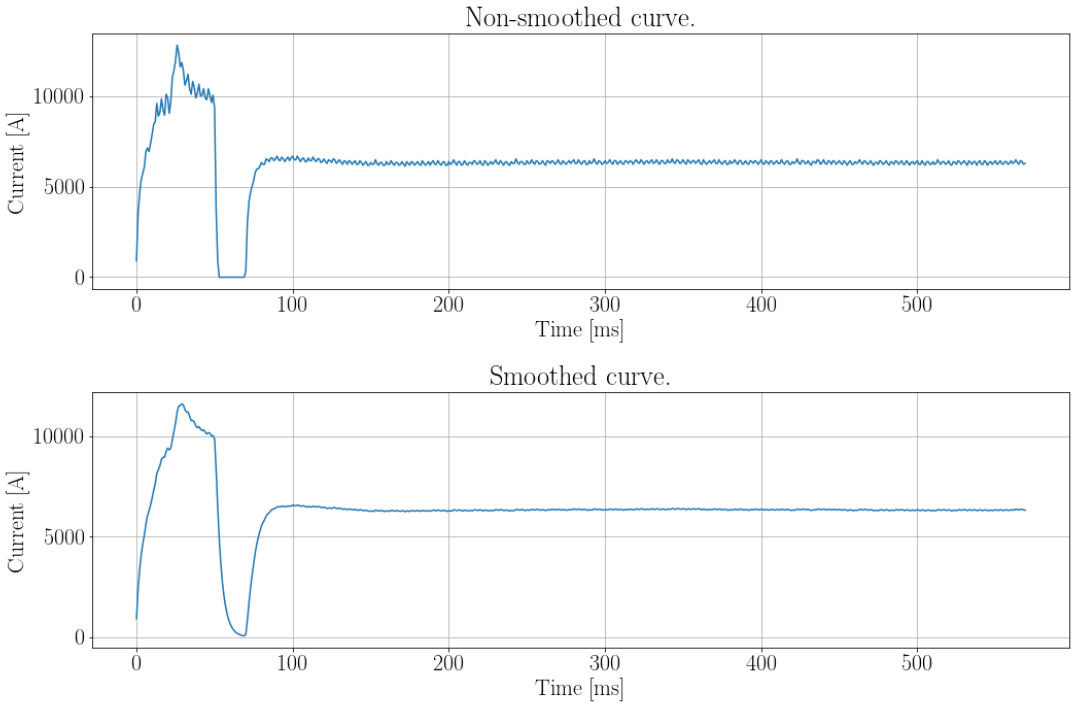


Figure 4.4: Comparison between the smoothed and non-smoothed version of the same current curve

on the other hand it is easy to select data with no meaning at all.
In this project, the first attempts done with the basic k-means algorithm have led to the extraction of meaningful statistical variables for the data under analysis, such as average values, medians, standard deviations and percentiles (this experiment is described in section 5.1). However, to better stick to the definition of time-series and to apply the already described time-series clustering (illustrated in section 2.2.3), there majority of the experiments has been conducted by selecting the actual curves as features, thus generating several matrices of features whose number of columns is equal to the number of points in the curve (or interval) under study. Figure 4.5 shows some examples of statistical measurements extracted from data (in this case the statistical measurements are extracted from data normalised following the procedure described in section 4.4).

4.4 Normalisation

Features extraction can be linked to the transformation phase of the KDD process, since it is a sort of mapping of the data into several features. Together with it,

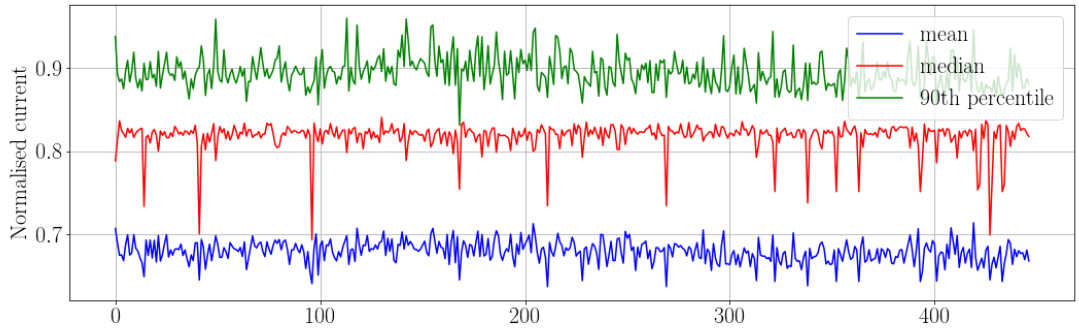


Figure 4.5: Example of statistical measurements extracted from data.

also the normalisation process can be included in the transformation phase. This task is very important, since it allows to avoid dealing with values that can bias the actual data mining procedures due to too high or too low values. Figure 4.6 shows the normalised version of an already presented set of curves. Actually, nothing changes from the graphical point of view, but it is clear that now the y-axis never goes below 0 or above 1. The normalisation technique chosen for this project is pretty simple, but efficient and it is the so-called "maximum-minimum" normalisation. This technique uses the maximum and minimum values among those to be normalised in order to bring any value between 0 and 1. Each value, then, is transformed using the following formula:

$$x_{norm} = \frac{x - min_i}{max_i - min_i} \tag{4.2}$$

in which x_{norm} is the normalised value, min_i and max_i are the minimum and maximum values in the dataset and x is the value to be normalised.

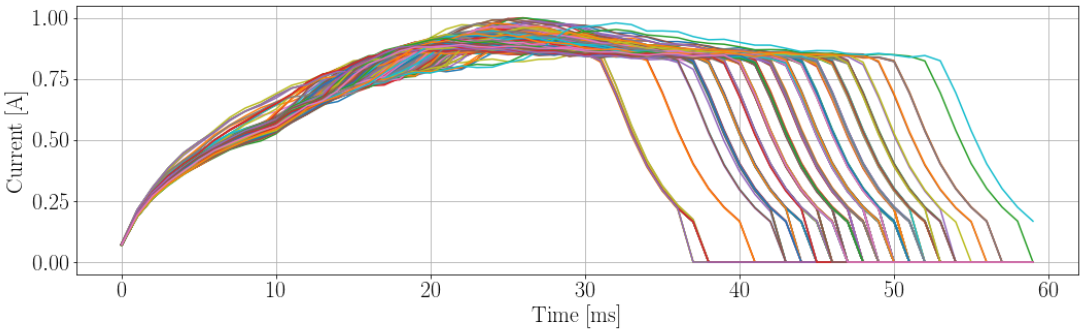


Figure 4.6: Normalisation example.

4.5 Intervals definition/Windowing

The preprocessing and transformation process does not end with normalisation, since there are two further steps to take; the first one is windowing. In section 3.3, the two welding techniques that have been encountered in this project are described. The main thing to notice about both the techniques is that the curves present at least two different behaviours throughout a welding process and this should be taken into consideration when performing clustering or other kinds of analysis. To counteract the presence of many behaviours inside a single process, the idea is to apply some windowing to the curves, in order to isolate their different parts and to analyse them separately. To demonstrate how useful it is to apply some windowing processes to the curves, let's make an example: if the idea is to perform some basic k-means clustering on the curves, using as features some relevant statistical parameters of them (such as mean value, median, percentiles...) it is clear that, for instance, the average of the first part of a curve characterised by the first welding technique (the three-step one) will be different from that of the third part (which is the steady state one) and then it will be difficult to consider the overall average of that curve as a reliable feature for the clustering process. Therefore, it is crucial to divide the curves in sub-intervals in order to avoid dealing with data for which it is simply pointless to perform an analysis of all of them together. This project presents two main types of windowing:

- a windowing process based on time, in which curves are divided in intervals depending on the time at which some relevant points are reached. For instance, considering the three-step welding technique, the idea is to get the 90^{th} percentile of the time instants (for all the curves under study in that specific experiment) at which the maximum (for the first part) and the minimum (for the second and third part) values of current (or voltage) are reached and to divide the curves themselves according to these two points. This process has the advantage of producing intervals of the same length (for the same type of interval), but for sure some of them will also include some points which are after the limit and some other will instead be "incomplete" since they will be cut before reaching the limit itself. This process is described in figure 4.7, which represents the same curve as figure 4.4 (the one after the filter application);

- a windowing process based on a threshold (of current, for example), in which when a certain curve reaches that threshold, an interval is generated. The main advantage is to have intervals featuring the same behaviour and profile, but on the other hand there will be the need of performing some alignment on the intervals themselves in order to avoid having different lengths for them (the alignment will be better described in section 4.6). Figure 4.8 shows an

example, always based on the three-step welding technique (in this case, the intervals are not aligned yet and moreover the second part is not considered as it is characterised by many values equal to zero).

Between the two windowing processes, the latter is the one that has been preferred as it is better to do something for the alignment of the curves rather than having intervals of the same length but with more or less points with respect to what is expected.

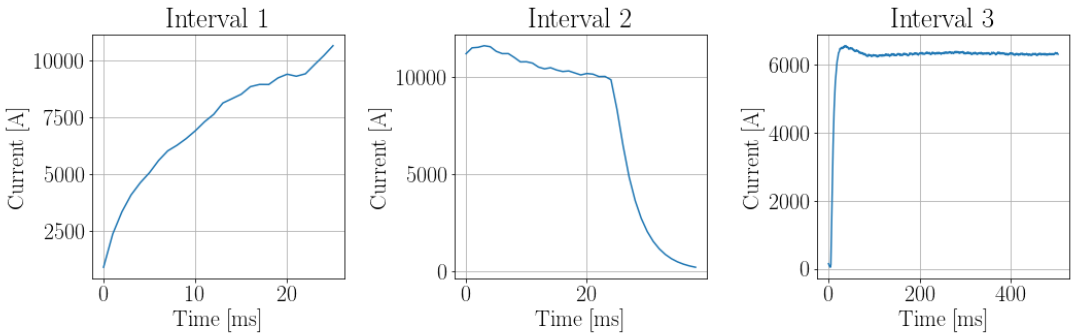


Figure 4.7: Example of the first windowing technique.

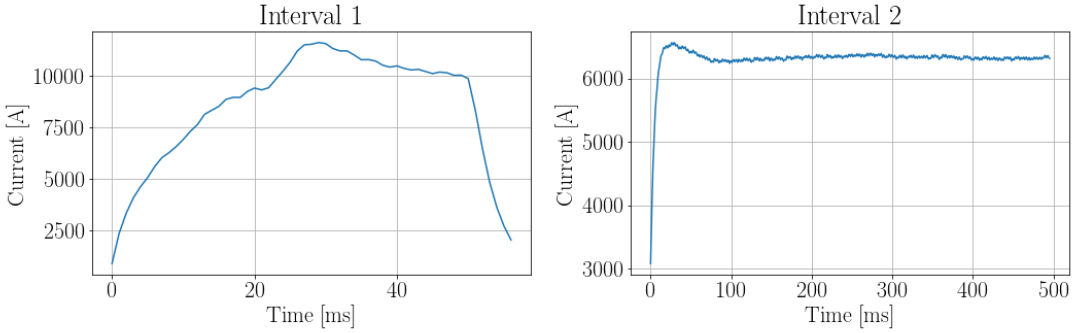


Figure 4.8: Example of the second windowing technique. It is clear that here, since it is a three-steps welding process, the stopping phase is not considered.

4.6 Alignment

As written in section 4.5, the second technique (the one based on the current or voltage threshold and not on the time) is the one preferred between the two that

have been described. However, it has a big disadvantage,as it provides intervals of different lengths, which therefore have to be fixed. To do that, there are many ways (for instance, it could be a good idea to use the Distance Time Warping together with a clustering algorithm) and in this particular case it is preferred to choose one of the simplest ideas. Considering the intervals in which a curve is divided (no matter what welding technique is used), the last one (i.e. the interval whose end corresponds to the end of the curve) is cut at the minimum length among all the last intervals under analysis, while the other ones (for instance the first one in the just mentioned two-intervals division) are instead extended until they reach the maximum length among themselves, adding some zeros at their tails. In this way, the length of each interval is always the same with respect to the ones in the same group; therefore, an easy solution is implied and moreover there are not so many issues with, for instance, the application of clustering algorithms. An example of this type of alignment is provided in figure 4.9, in which it is clear how the curves of the first interval arrive all to the same time instant with different amounts of zeros and how the ones of the second group end all at the same instant.

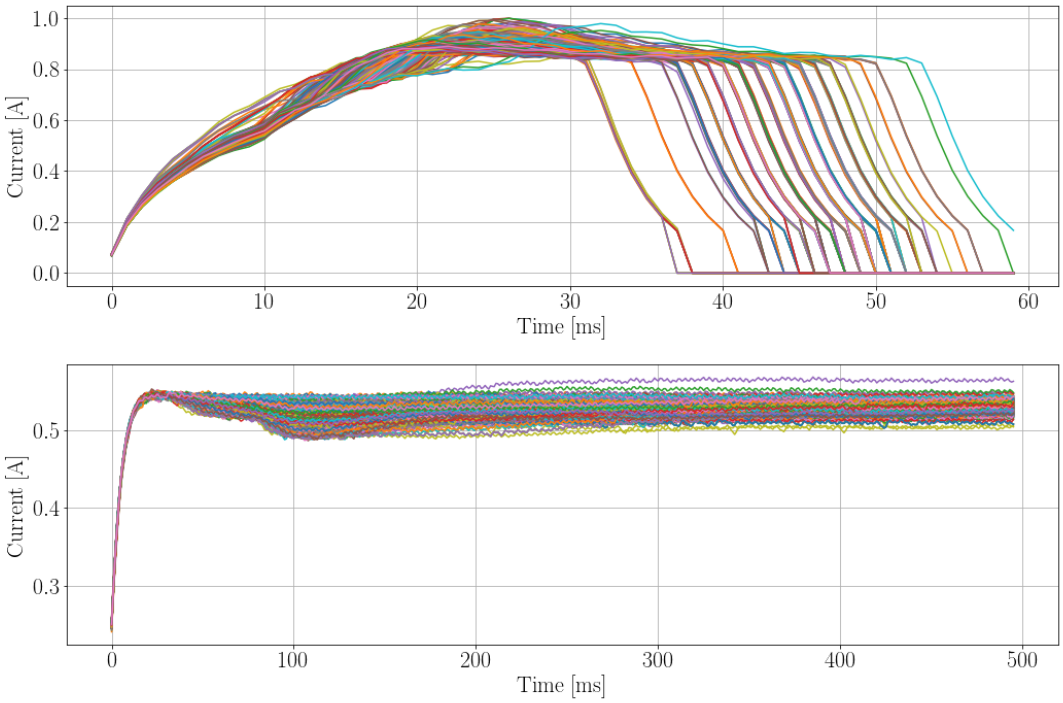


Figure 4.9: Example of the alignment method on two different interval of the same curve.

4.7 Clustering

The process of data selection, cleaning and preprocessing is actually quite long, but then the part in which results are provided starts with clustering. The clustering process has already been described in sections 2.2 and 2.2.3. As written before, clustering means grouping together in sets (or clusters) object coming from the same dataset that are linked or correlated in some way. Therefore, it can be said that this is one of the first steps towards the definition of patterns and therefore towards the acquisition of knowledge about the data under analysis.

4.8 Clustering evaluation

Clustering should not be just executed without control, but some evaluation would be needed. To evaluate how good a clustering process is, in this project the silhouette coefficient computation is chosen among the different methods described in section 2.2.4. In this way, it is possible to determine how good are clusters formed with k (number of clusters), recalling that the number of k with the highest value of silhouette score is then picked up for the experiment.

4.9 Correlation with metadata

The last part of this working pipeline is actually pretty fundamental, since it determines if it is possible to define relevant patterns and therefore get some meaningful knowledge about the analysed data. This can be done in many ways, but the easiest and most efficient one is the correlation of data (if available) with some important metadata present in the dataset. One of the clearest examples, also related to the project presented in this work, is the correlation of voltage or current curves against the value of TDC, labelling them with the cluster they are assigned to by the executed algorithm, hoping to find some patterns that can start the predictive maintenance schedules definition.

Chapter 5

Results

5.1 K-means clustering with statistical measurements as features.

The very first experiment that has been performed is the application of a standard k-means clustering algorithm to a specific set of voltage and current curves which is the one that is also described and analysed with the custom algorithm in section 5.3.2. Those curves are related to a specific spot (described in section 3.4.2) whose peculiarity is that of being (according to the provided technical specifications) the first one after each dressing or electrode replacement, therefore it is the best spot for trying to put on evidence possible differences between welds executed by consumed electrode or by "fresh" ones. As said, the algorithm run in this case is the standard k-means, because the features implied for the clustering process are not the time-series, but single data points representing various meaningful statistical features (as mentioned in section 4.3) extracted from the curves themselves.

In section 4.3 and in particular in figure 4.5, some examples of statistical features extracted from the time-series are provided. In the figure, the average, the median and the 90^{th} percentile extracted from the current time-series of the already mentioned spot are shown, as they are used also in the following example. It is important to notice that the features are extracted from the current curves because the current is the one that drives the welding processes. Apart from the average values, which is chosen as it is the most common and used operator, the median (that can be also addressed as the 50^{th} percentile) and the 90^{th} percentile are extracted in order to provide more robust operators with respect to the mean value itself (recall that the average is much more sensitive to outliers with respect to percentiles and in particular with respect to the median, that is less biased by "weird" values).

Figure 5.1 shows the current curves under analysis. Since they are characterised by a three-steps welding technique, the first windowing technique is applied, therefore the one based on time instants and maximum/minimum values of the time-series (the first one of the two presented in section 4.5). In this example, only the values related to the first phase (then the one between the beginning of each time-series and the time instant at which the maximum is reached) are going to be used to extract the already mentioned statistical features. The division and the separate extraction are necessary because curves show very different behaviours within themselves and each one of the three phases is different with respect to the others. First-interval current curves under analysis are shown in figure 5.2.

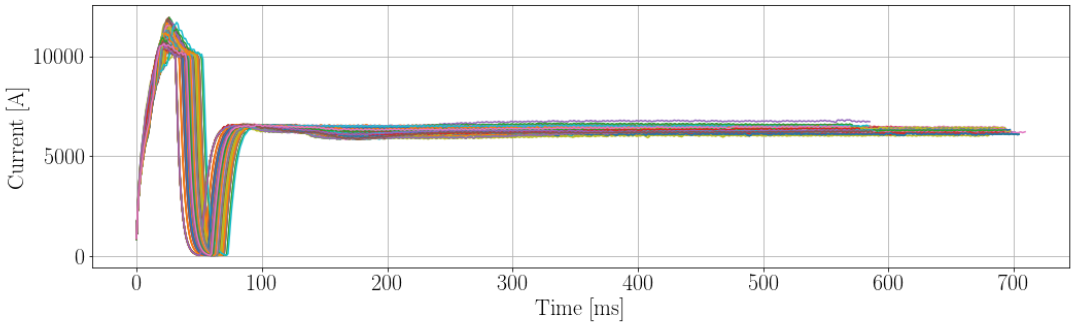


Figure 5.1: Smoothed current time-series from which statistical features are extracted.

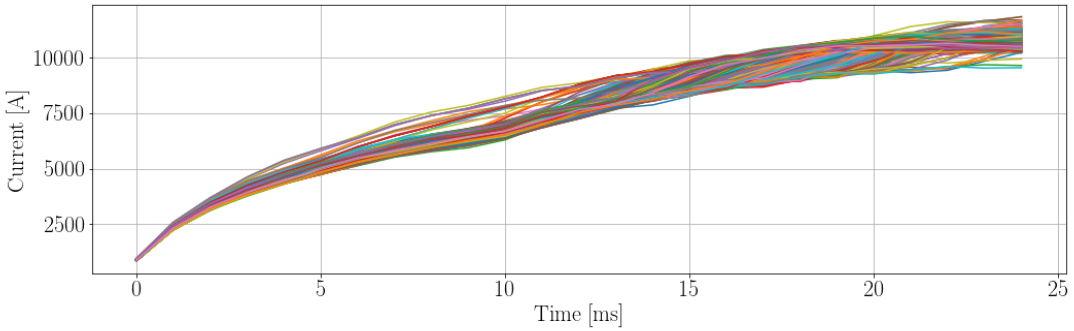


Figure 5.2: First-interval current time-series derived from the application of windowing to the curves in figure 5.1.

After the definition of the interval from which the features are extracted and after the extraction of the features themselves, it is now time to run the k-means

algorithm. The first thing to do is the evaluation of the best number of clusters and in figure 5.3 it is possible to see that the optimal k in this case is two, since the related silhouette score is much higher than any other one and moreover it is far above 0.50.

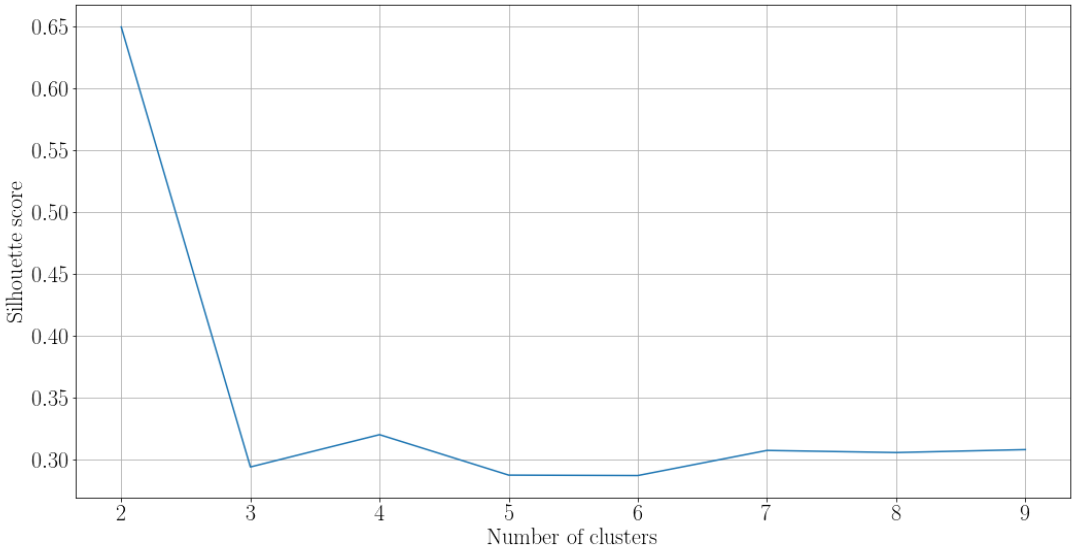


Figure 5.3: Silhouette score evaluation for the k-means algorithm with statistical features.

The second thing to do is the evaluation of the clusters and in particular the analysis of the correlation with the metadata, which in this case means studying the behaviour of the cluster labels in time with respect to the TDC. This is represented in figure 5.4 and unfortunately the situation is not extremely good, since not only there is not a periodicity, but the division in clusters is actually really poor, as most of the data points are put in one of the clusters and only a very small group is labelled with the other one. This puts on evidence the difficulties that arise when the features extraction is not suitable. In fact, as said before, in this case it is not very useful to extract statistical features from the time-series (even if they are divided in different intervals) because the behaviour of the curve is not represented by a single number. For instance, two curves could be extremely different with respect to each other in terms of trend and values, but they could still share the same average value. This is why it is better to perform the clustering directly on the time-series, which therefore become the actual features.

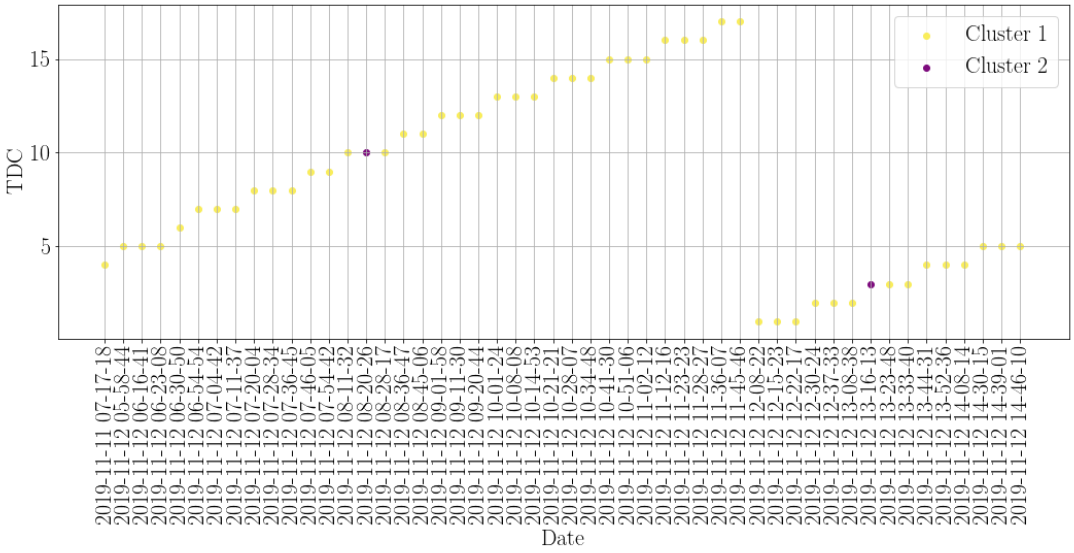


Figure 5.4: Correlation of standard k-means clusters with TDC.

5.2 Custom-distance algorithm application on a synthetic data set

First of all, the custom-distance clustering algorithm is tested on a set of synthetic data that has been generated with two main purposes:

- demonstrate that the algorithm is feasible and provides satisfying results when executed in a trivial situation;
- compare the algorithm itself with a standard k-means, to put on evidence how the latter is not suitable for time-series clustering.

The synthetic data set is based on two base curves:

- a simple sine $y(t) = sin(4\pi t)$;
- a simple cosine $y(t) = cos(4\pi t)$.

To both the curves a random Gaussian noise $X \sim N(0, 0.1)$ is added (even if there are also some attempts done with a variable variance σ^2). The two base curves are then combined to create four families of curves that actually emulate the voltage/current couples of curves present for the real provided data set. In particular, the four synthetic families of curves are:

- $sin(4\pi t)$ and $cos(4\pi t)$;

- $cos(4\pi t)$ and $sin(4\pi t)$;
- $sin(4\pi t)$ and $sin(4\pi t)$;
- $cos(4\pi t)$ and $cos(4\pi t)$.

In figure 5.5 the four families of curves are represented.

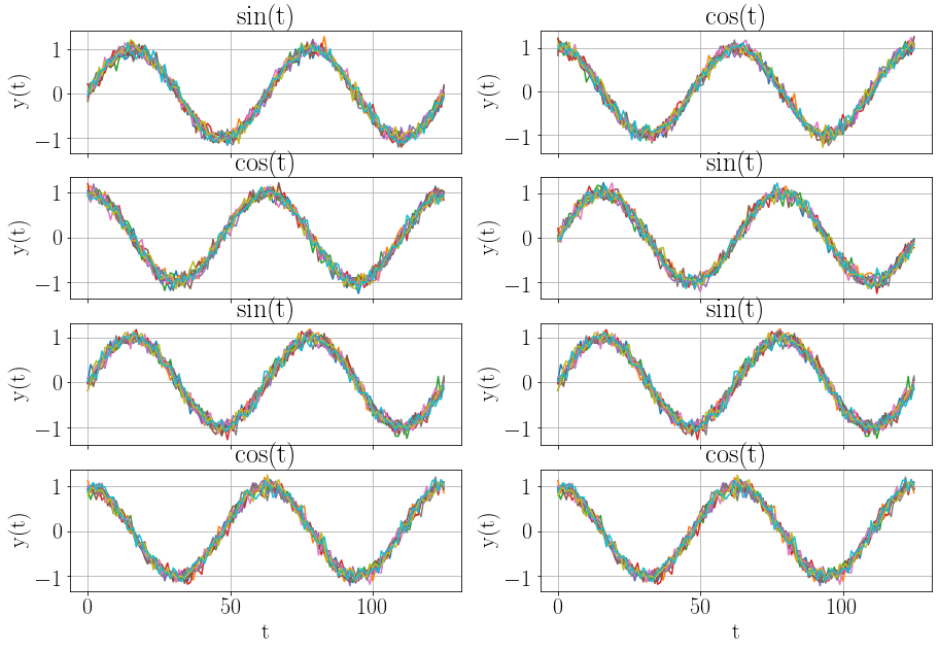


Figure 5.5: Curves of the synthetic data set, divided by family.

Before applying the custom-distance clustering algorithm to the synthetic data set, let's analyse the results obtained with k-means algorithm. In figure 5.6, it is possible to see how the silhouette score behaves when the number of clusters varies and how the centroids are set with the optimal number of clusters, which in this case is two (and this is expected since there are only two base curves). It is important to remember that the application of k-means can be performed on the synthetic data set only by considering the curves separately, otherwise it would be difficult to obtain a good result. Figure 5.7, then, shows the clustering results for k-means on the synthetic data set, and once again it is possible to notice the good job done by the algorithm when considering the two curves in a separate way.

It is now time to apply the custom-distance algorithm to the synthetic data set. The first thing to do is to perform a validation of the algorithm and this is done

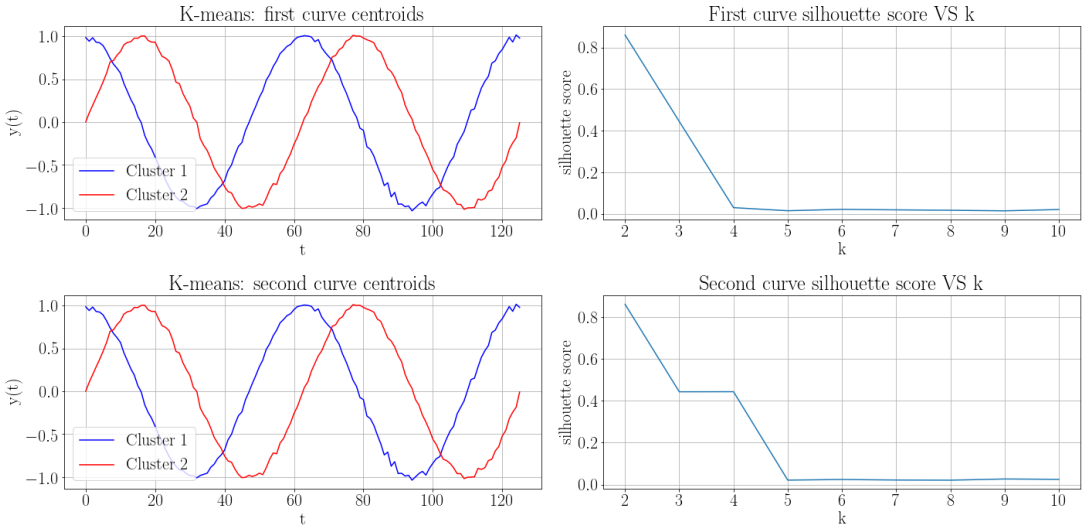


Figure 5.6: Silhouette behaviour and centroids with optimal number of clusters for k-means application on the synthetic data set.

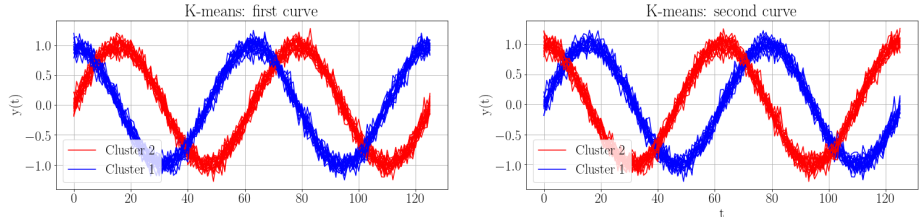


Figure 5.7: K-means results for synthetic data set with optimal number of clusters.

by comparing the results obtained with the custom-made silhouette function and those provided by the one developed inside the scikit-learn Python library [11], which also contains the code of the just executed standard k-means algorithm. In figure, 5.8 it is possible to see how the two functions behave in the same way, thus it is possible to say that the custom algorithm is validated in terms of silhouette function.

After the validation of the silhouette function, the silhouette score itself is computed for different number of clusters, in order to see if the algorithm chooses the ideal value (which in this case is four, since there are four families of curves in this synthetic data set). Figure 5.9 shows that the number of clusters with the highest silhouette score is exactly four, as expected, and then it is possible to run the algorithm with this value and see the results it provides. In figure 5.10 the results of the custom clustering algorithm run on the synthetic data set tell that the

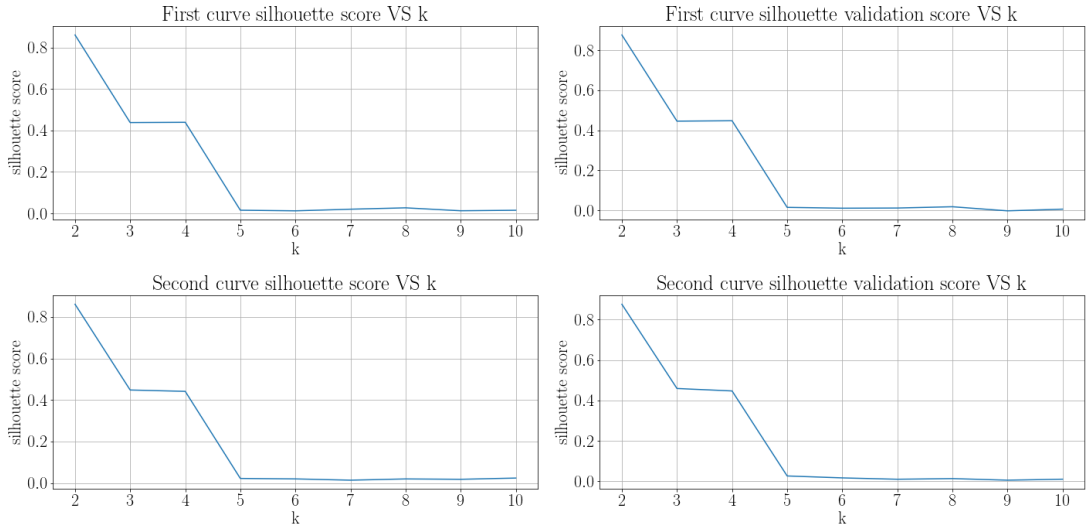


Figure 5.8: Custom silhouette function comparison with the one developed in the scikit-learn Python library.

algorithm itself works well and therefore it can be performed over real data coming from welding measurements done at the plant.

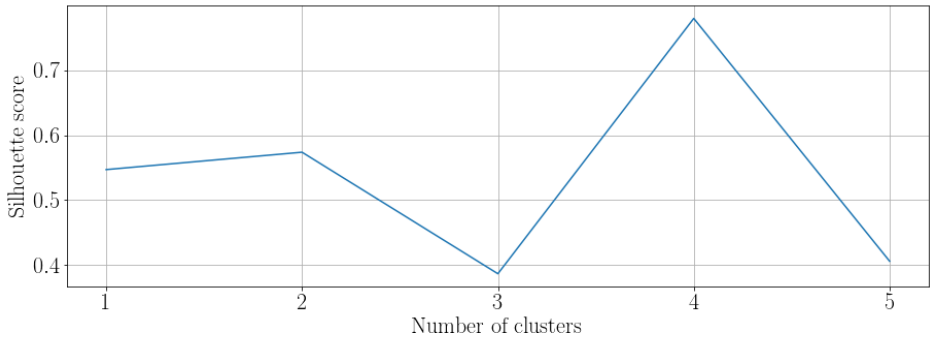


Figure 5.9: Silhouette score VS number of clusters for the synthetic data set.

To ensure the good functioning of the algorithm, a second test can be done with the synthetic data set. The idea is to increase the variance of the Gaussian noise added to the sinewaves to understand if the algorithm remains able to distinguish the four family curves. In figure 5.11, the clustering results of this test are presented (the number of clusters is set to four as before) and it is possible to say that the algorithm works well also in this case (even if the noise is much more present), since the division in clusters reflects the four different families of curves.

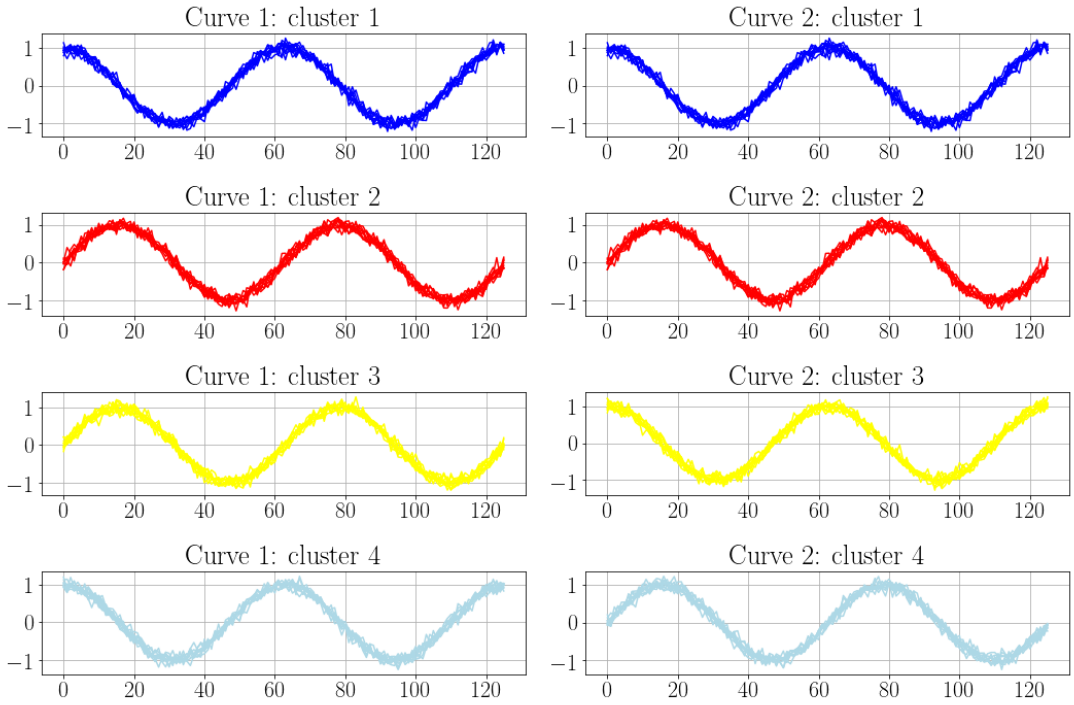


Figure 5.10: Custom clustering algorithm results on the synthetic data set.

5.3 Predictive maintenance

After having demonstrated that the custom algorithm works well by applying it to the synthetic data, it is possible to start the analysis on the real ones provided by the measurement tools mounted on the welding robots.

5.3.1 Adjustment welds

The first experiment is about the adjustment welds, that (as described in section 5.3.1) are welding processes not executed on a spot and performed after every electrode replacement just to make sure that everything works and to reset the position of the electrodes themselves. Figure 5.12 shows the current and voltage curves related to the welding processes under analysis: it is clear that the welds are much shorter in time than those on the spots, since they last about 100 ms. It is important to underline that the curves presented in this figure have already been smoothed with the exponential moving average filter.

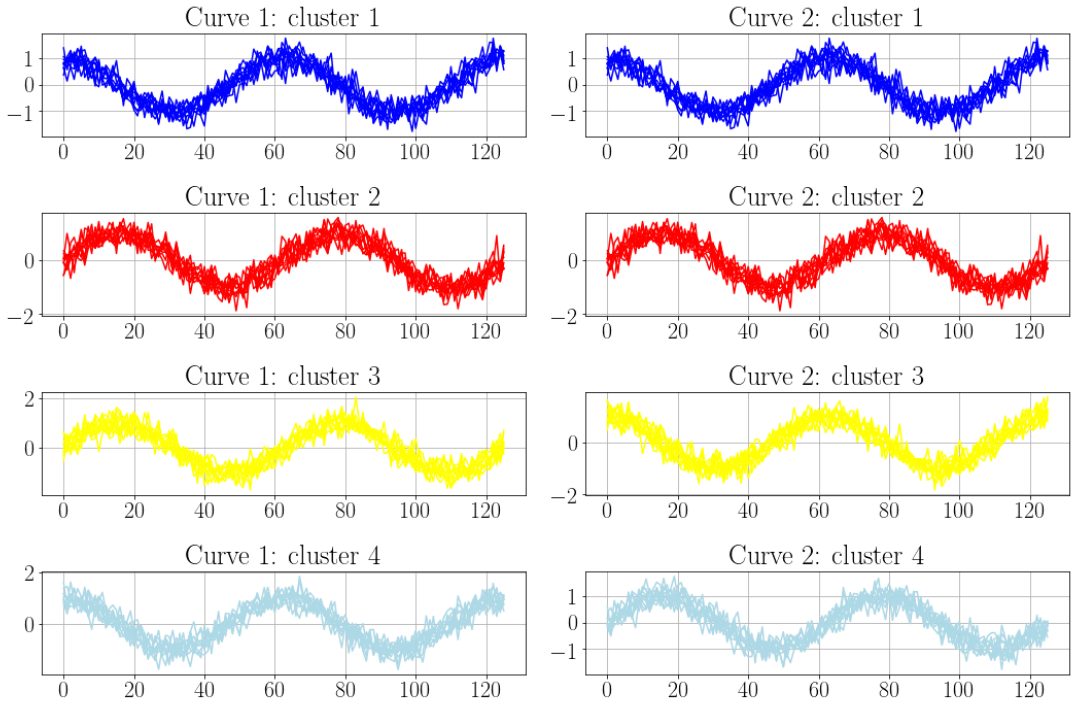


Figure 5.11: Custom clustering on synthetic data set with increased variance of the Gaussian noise.

Recalling that the clustering algorithm is executed on normalised data, figure 5.13 shows the silhouette scores obtained with different values for the number of clusters. The highest value is obtained with 2 clusters, therefore k is set to 2.

Before evaluating the results, it is important to underline that the curves related to adjustment welds are not subject to windowing since the welding time is not so long. Figure 5.14 shows the results of the custom-distance clustering algorithm applied to the data under analysis in this section. To read and give a meaning to the presented results, they should be evaluated by correlating them to relevant metadata. In this case, the division in clusters is correlated with the data of the TDC, in order to understand if it is possible to recognise a pattern or a particular behaviour. Figure 5.15 shows the TDC of a portion of all the files related to the spot under analysis (they are presented in chronological order, with date and time of the welding process on the x axis), but each point is labelled with a color corresponding to the cluster to which the related WeldLog file has been assigned. Unfortunately, there are no evident patterns since the colors do not repeat themselves following a precise scheme but actually appears randomly. The lack of a pattern suggests

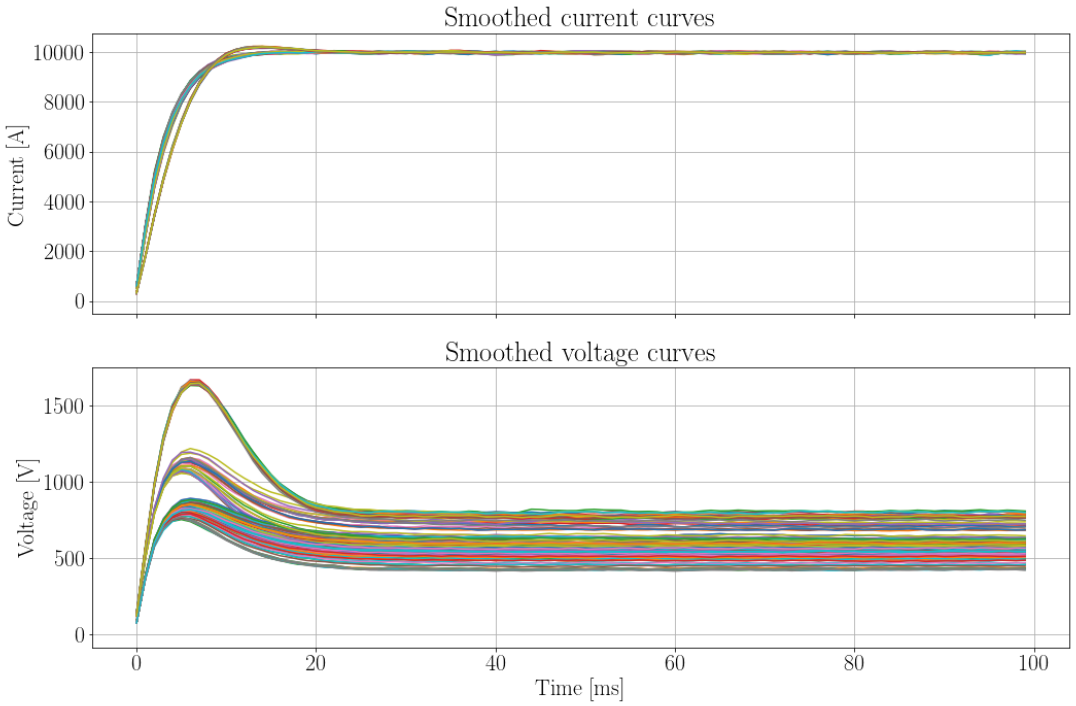


Figure 5.12: Voltage and current curves coming from adjustment welding processes.

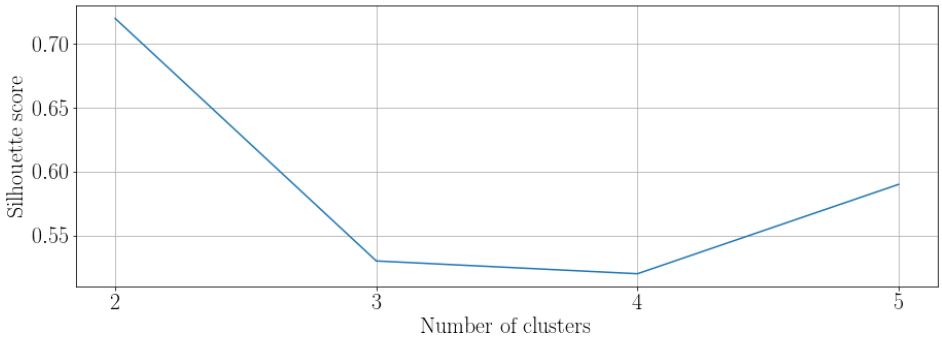


Figure 5.13: Silhouette score VS number of clusters for adjustment welds experiment

that there is no correlation between the clusters and the metadata and this puts on evidence the importance of evaluating the clustering results with the metadata themselves.

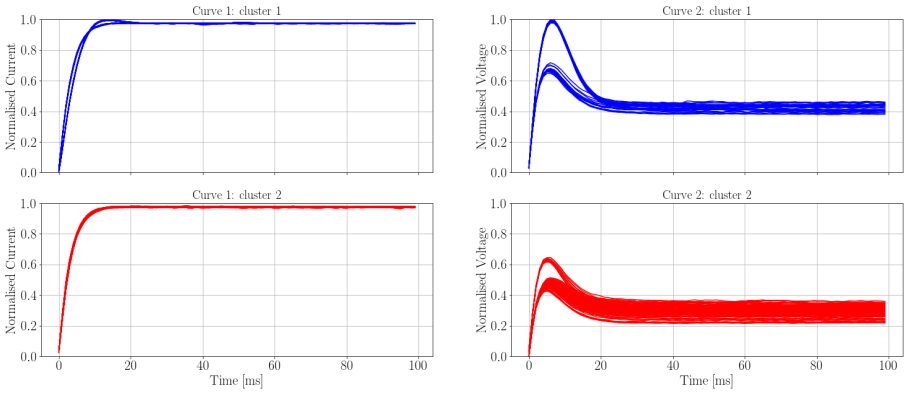


Figure 5.14: Results of the custom algorithm application on the adjustment welding files.

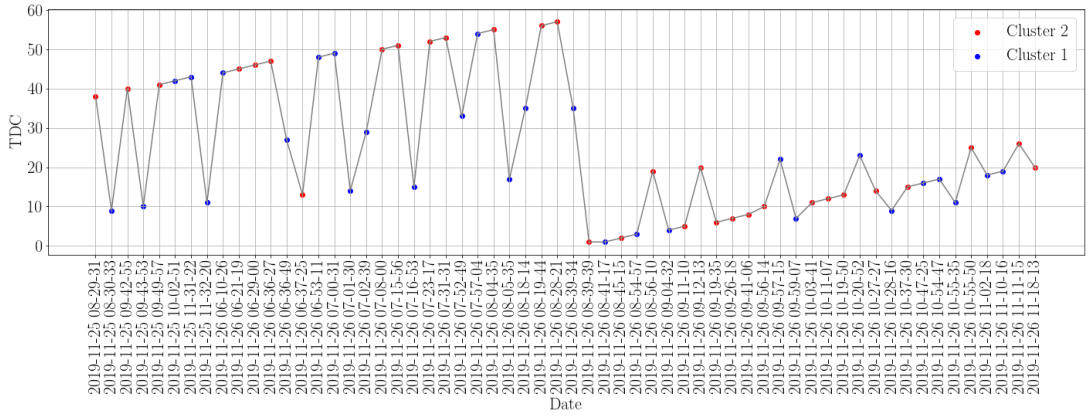


Figure 5.15: TDC data for the adjustment welds, labelled by cluster.

5.3.2 Unsupervised clustering

The second part of the experiments related to predictive maintenance regards actual spots on which the custom clustering algorithm is applied. Among all the spots available, the one chosen is particular, since (as already said in section 3.4.2) it the first point after a dressing process and therefore it is the one that always starts a welding sequence after a dressing itself or an electrode replacement.

As shown in figure 5.16 (which represents the smoothed voltage and current curves for the spot under analysis), the welding technique that characterises this spot is the one with three phases and due to that the clustering algorithm is applied after

a windowing process that divides each curve into three parts, of which the first (from the beginning of the welding process until the moment in which the current goes below a threshold of 2000 A) and the third (from the end of the stopping phase until the end of the weld) are picked. The decision to exclude the second part of the weld, which is the stopping one, is due to the fact that current and voltage levels are null for most of the time during this phase and then there is no interest in executing clustering analysis on them.

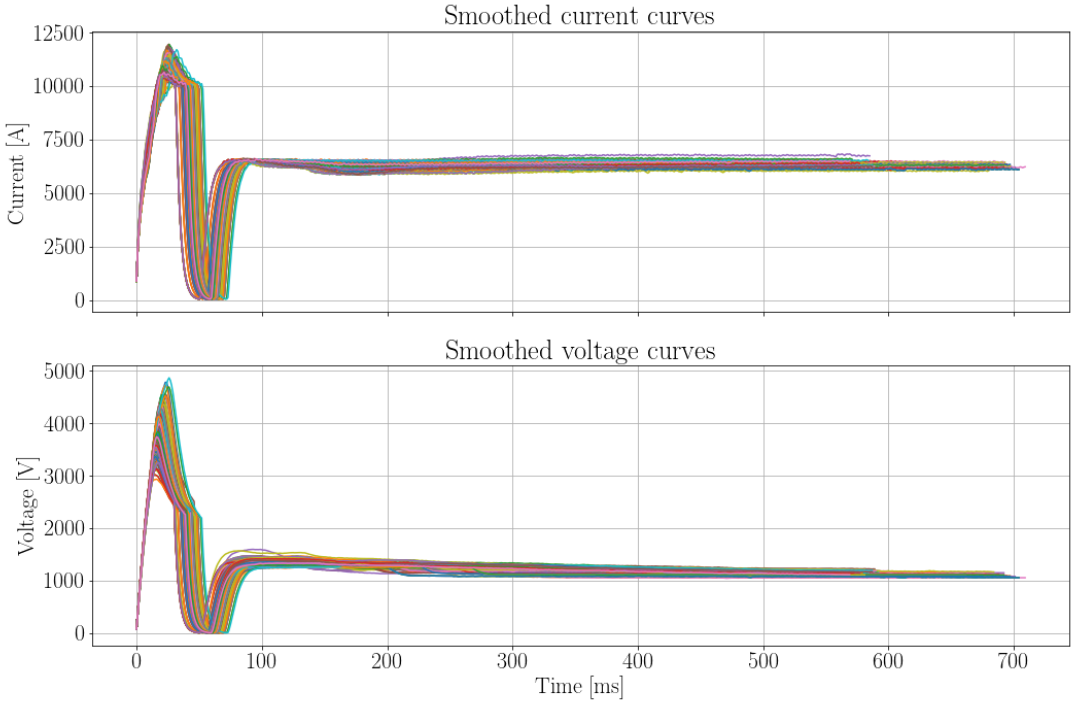


Figure 5.16: Voltage and current curves for the after-dressing spot.

As it has been done in section 5.3.1 , it is now time to evaluate the performance of the custom-distance clustering algorithm on the spot files analysed in this section, correlating them with the metadata of TDC. In this particular case, the already mentioned windowing process leads to the performance of two clustering experiments, one per interval.

For both the intervals, curves are smoothed, normalised and above all aligned to each other in the following way:

- first-interval curves are extended by adding zeros at the end of each of them

until any curve has the same length with respect to the others (the length is the maximum one among all the curves);

- second-interval curves are instead cut at the time instant such that any curve has a length equal to the minimum value among all of them.

Figure 5.17 shows the behaviour of the silhouette score with respect to the number of clusters. The highest value coincides with k set to four and it is actually pretty decent since it is slightly above 0.50; this indicates that the clustering algorithm has performed well.

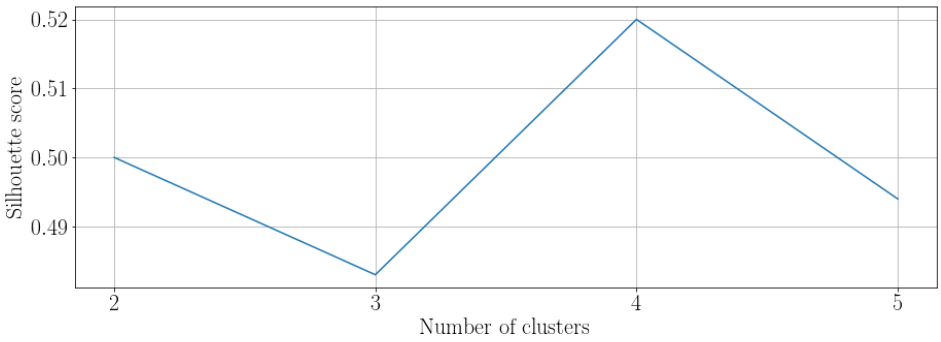


Figure 5.17: Silhouette score VS number of clusters for the first-interval curves of the spot analysed during the unsupervised-clustering tests.

After having observed the silhouette score's trend, it is now time to evaluate the results of the clustering process performed with the properly set number of clusters. Those results are shown in figure 5.18 (current curves on the left, voltage ones on the right), from which it is possible to deduce that the clusters are pretty different from each other, thus confirming the outcome of the analysis of the silhouette score (as it suggests a good clustering performance provided by the algorithm).
It is possible to go even further with the evaluation of those outcomes by studying the correlation of the clusters with the TDC (as already done in the previous experiment) and also trying to spot eventual patterns by observing the sequence of the clusters themselves with respect to the date and time of the welding processes. For what concerns the correlation with the TDC, figure 5.19 clearly demonstrates that there is no evident pattern in the sequence of cluster labels and TDC values and this is also proved by figure 5.20, which instead shows only the sequence of clusters VS time of the weld, once again not presenting any recurrent behaviour.
Looking at the second-interval curves, the situation is a bit trickier, since being in the steady-state part makes all the curves quite similar to each other. In fact, as shown in figure 5.21 (in which normalised and aligned second-interval voltage and

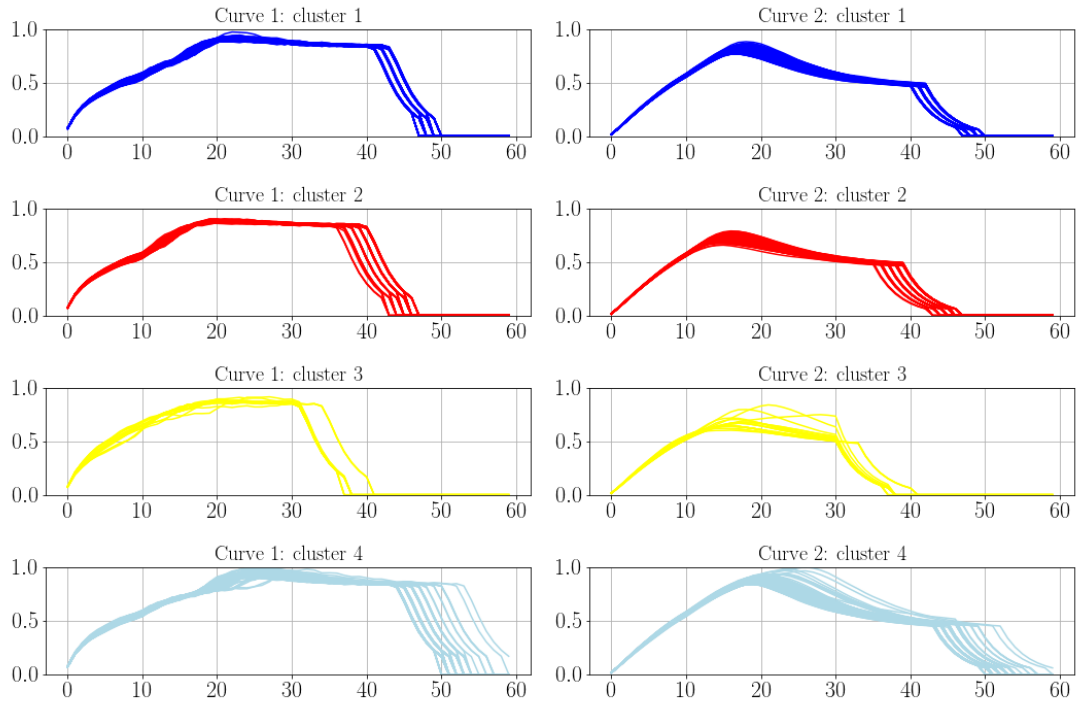


Figure 5.18: Division in clusters for the first interval curves of the after-dressing spot.

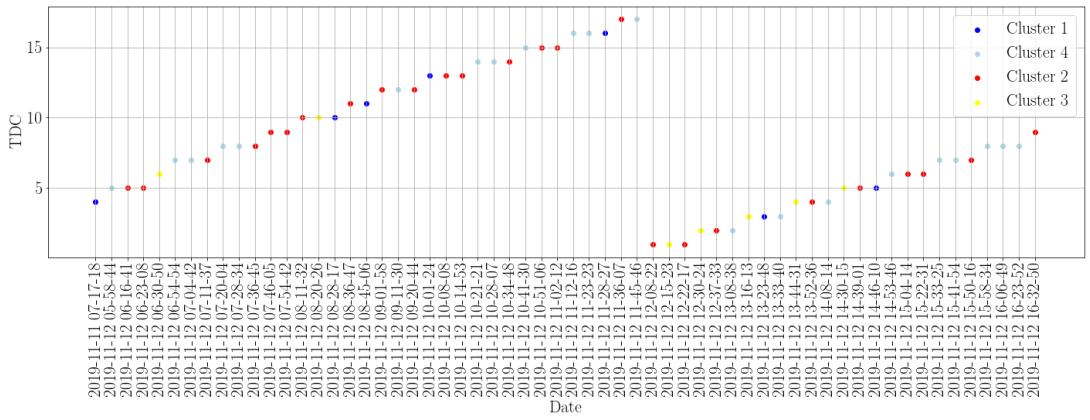


Figure 5.19: First interval after-dressing spot curves: correlation with TDC.

current curves are displayed), each curve presents little differences with respect to the other ones.

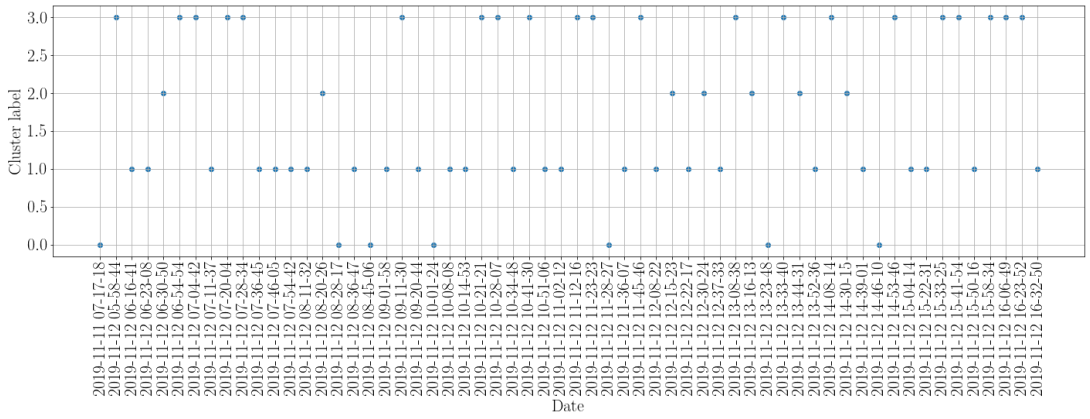


Figure 5.20: Sequence of clusters with respect to welding time for the first-interval curves of the unsupervised analysis.

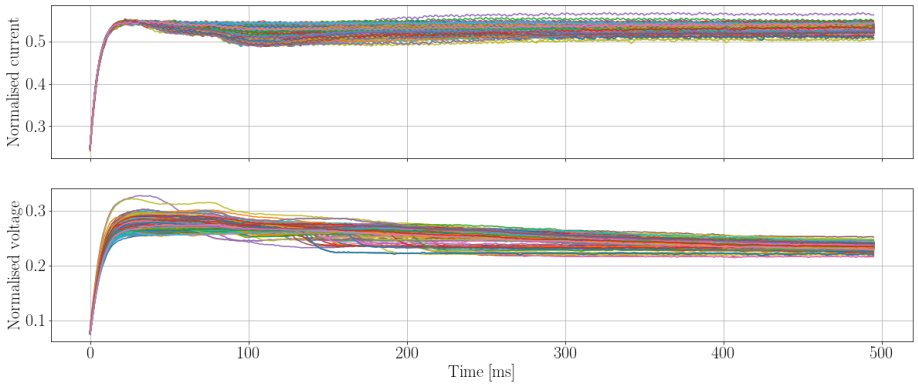


Figure 5.21: Second-interval normalised and aligned curves.

In figure 5.22, the trend of the silhouette score with respect to the number of clusters is shown for the just mentioned second-interval curves. The fact that the highest silhouette score is obtained with k=2 confirms the greater difficulty in dividing curves into groups, compared to the case of the first-interval ones. However, as already said, this was expected since the steady-state behaviour of any curve is pretty similar to that of any other one and therefore the division is cumbersome in this situation.

It is now possible to consider the outcomes of the clustering algorithm performed on those curves. Results are shown in figure 5.23, in which, as before, voltage curves are on the right and current ones on the left. The high similarity among all

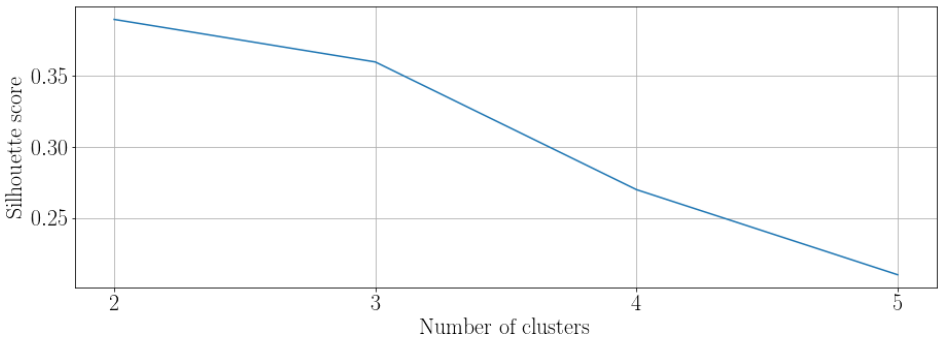


Figure 5.22: Silhouette score for second-interval curves studied during unsupervised analysis.

the curves is put on evidence once again and in fact the silhouette score values are all far below 0.50.

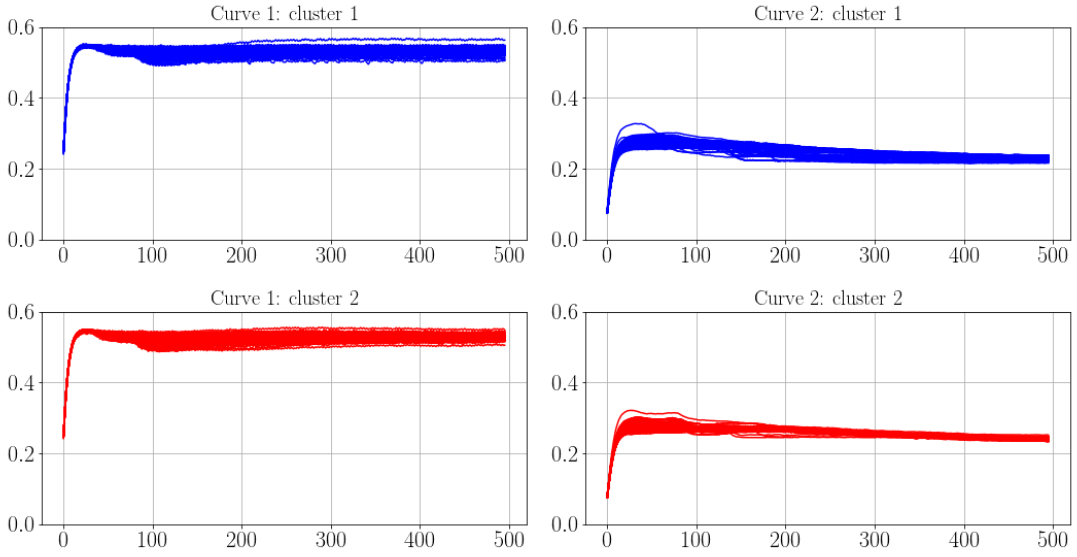


Figure 5.23: Division in clusters for the second interval curves of the after-dressing spot.

The evaluation of the correlation between clusters and metadata is presented in figure 5.24, in which once again no relevant pattern or particular behaviour can be spot. This is confirmed by figure 5.25, in which the sequence of clusters with respect to the time of the welding process is shown and in which no periodicity or

pattern are present.

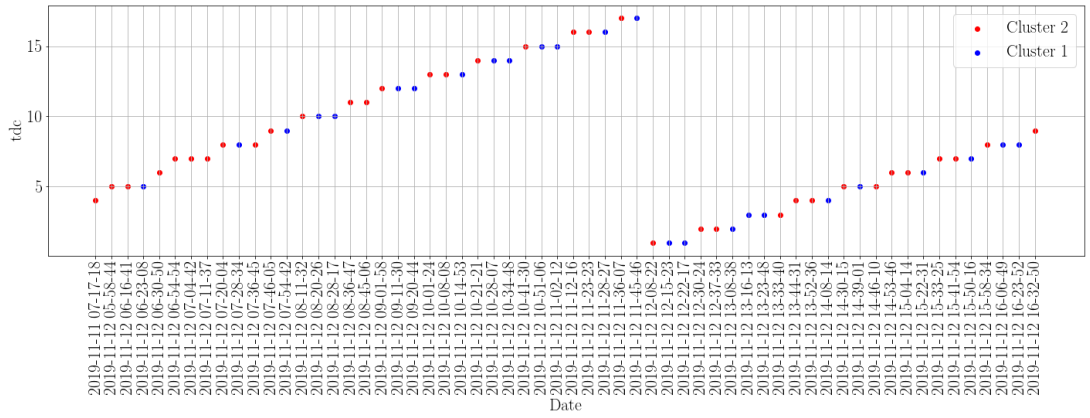


Figure 5.24: Second interval after-dressing spot curves: correlation with TDC.

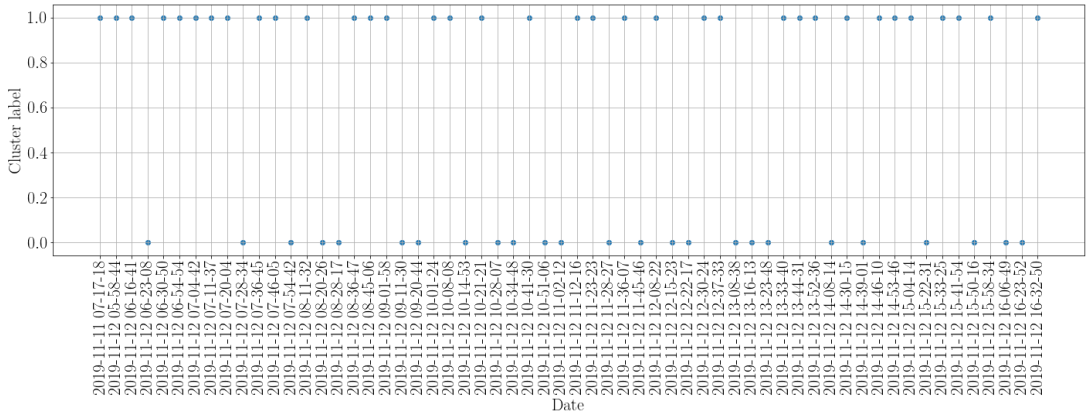


Figure 5.25: Cluster sequence for second-interval curves evaluated in the unsupervised-clustering experiment.

5.3.3 Aggregation and supervised analysis

Another interesting experiment that has been performed regards the so-called "supervised analysis". While 5.3.2 presents the attempt to group together similar curves without knowing anything in advance about them, now it is time to perform this grouping task also exploiting some labels given by the analysis of the metadata that are provided together with the time-series. The considered metadata are the TDC (which is the distance, in terms of car bodies, from the last electrode

replacement) and the distance from the last dressing.

It is important to underline that, for this particular task, the curves have been considered in their totality and therefore no windowing has been performed, while the analysis has been led separately for the current and voltage curves, using a standard k-means silhouette function, instead of the already mentioned and described custom-distance one.

The main set of information that has been used is that referred to the TDC, whose knowledge is useful for two reasons:

- first of all, it can be used as an important label to distinguish welding processes made by an already consumed electrode from those performed by a new one;
- it can be used to determine the distance from the last dressing, combining the TDC value itself with some specification coming from the plant in which the welding processes take place.

In table 5.1 it is possible to see how small the silhouette values are for all the spots considered and for both the two kinds of features utilised (recall that the silhouette score is considered satisfying if it is at least equal to 0.50).

Spot	Features	Silhouette score
1	Current curves	0.0435
2	Current curves	0.0309
3	Current curves	0.0082
1	Voltage curves	0.0360
2	Voltage curves	0.0820
3	Voltage curves	0.1060

Table 5.1: Aggregation/Supervised analysis: silhouette score of groups formed considering TDC.

Table 5.2 confirms the absence of correlation between these groups and the metadata, since the silhouette scores are almost always very low, even if in two cases the value is higher than 0.50.

5.4 In-sequence analysis

The analysis already introduced in section 3.4.4 is different from the ones led so far, since it does not deal with clustering and correlation with metadata, but it

Spot	Features	Silhouette score
1	Current curves	0.4095
2	Current curves	0.6108
3	Current curves	0.1206
1	Voltage curves	0.3472
2	Voltage curves	0.5848
3	Voltage curves	0.2145

Table 5.2: Aggregation/Supervised analysis: silhouette score of groups formed considering distance from last dressing.

studies what happens inside the same sequence of welding spots in terms of average current and resistance modifications. Considering one of the robots working on the right side of the car body, the idea is to understand if some particular behaviours can be detected by evaluating the trend of the average current and the average resistance in time and in correlation with the TDC, performing the analysis on particular spots belonging to the same welding sequence ("welding sequence" means a sequence of points where the related welding processes are not interspersed with electrode changes or dressing). To do that, two spots are picked:

- the spot that is at the beginning of each sequence for the right side;
- the spot that is at the end of each sequence, again for the right side of the car body[1].

In figure 5.26 the first analysis is presented. Given the two spots, it is clear that the average current is totally independent with respect to the TDC, since there is no specific behaviour in time and above all no particular pattern or trend. The same happens in figure 5.27, in which the same analysis is led, but considering the average resistance instead of the average current. The only thing that can be said is that in general the sequence-starting spot features a higher average current and a lower average resistance with respect to the sequence-ending one, but this is not enough to call it "a significant behaviour". One last thing to notice is that the just presented analysis is led on current and resistance because the current is the size driving the welds, while the resistance (even if it is derived from current and voltage) has been addressed as very important for the evaluation of the mechanical process by the domain experts working on the project.

[1]it is no useful to consider the whole sequence, therefore the two sides, since they are related to different robots and clamps.

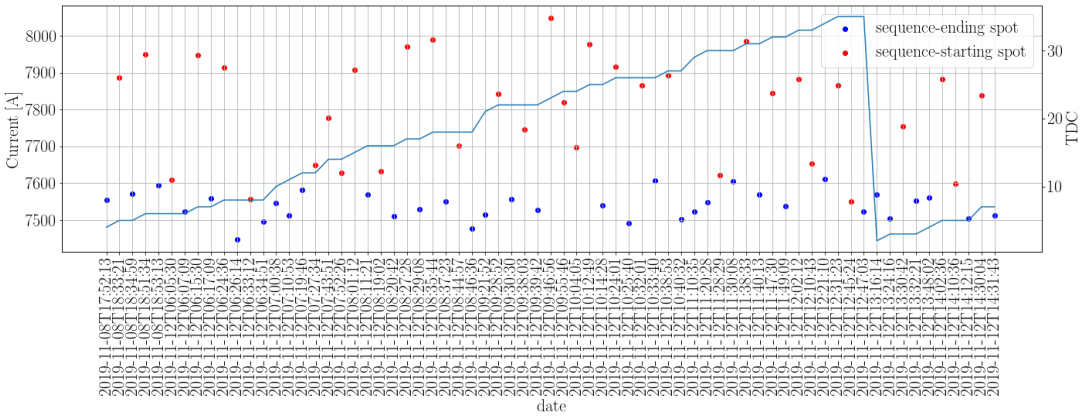


Figure 5.26: In-sequence analysis: average current VS TDC.

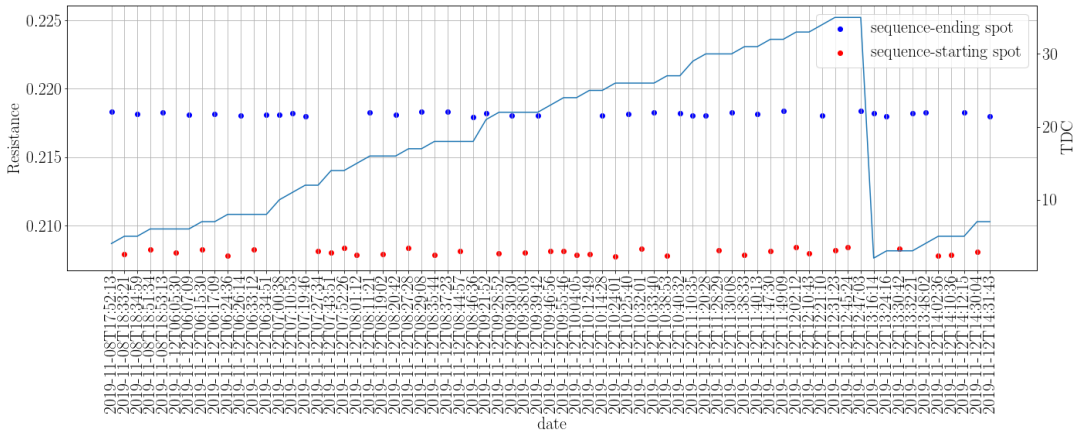


Figure 5.27: In-sequence analysis: average resistance VS TDC.

5.5 Anomalies

The analysis related to clustering and predictive maintenance suggests that, since there is no correlation between the clusters and the metadata, in the end a periodic maintenance scheme can be considered valid in this particular case. Anyway, apart from the clustering analysis, another kind of study has been led on the provided data and it regards faults.

It has already been said that, for the welding process under analysis, it is possible that a fault occurs. More specifically, an anomaly is defined as the unwanted expulsion of material from the welding site and it can happen for many different reason. The idea is to understand if the presence of a fault can influence somehow

the production quality or the energy consumption.

5.5.1 Preliminary analysis

First of all, let's start by focusing on the data to analyse. They all belong to the welding processes executed in a specific day, during which more than two complete sequences have been done, and above all they are all related to the same robot, since it makes no sense to analyse and compare data coming from different machinery.

Figure 5.28 shows the time instant at which each fault occurs of the data under analysis. It can be said that the majority of the faults is concentrated between 100 and 250 milliseconds, then just before or at the beginning of the steady-state phase (recall that it does not matter which welding technique is implied to the data, since there is always a steady-state phase).

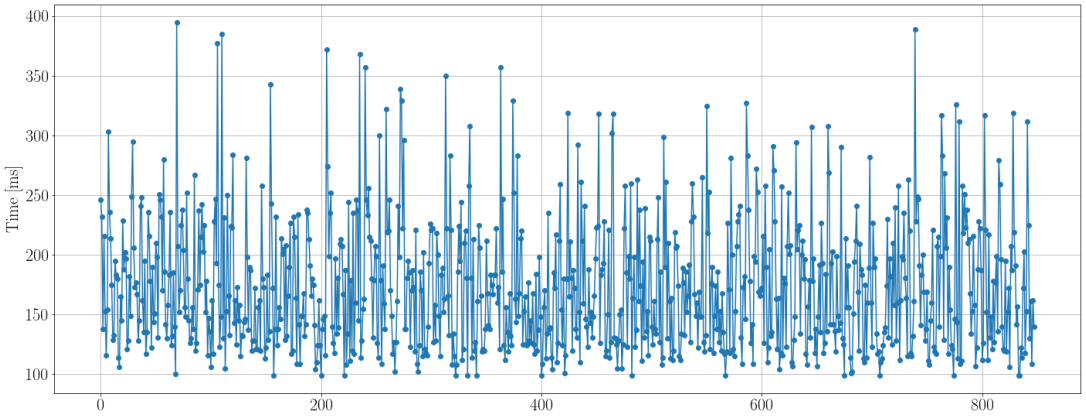


Figure 5.28: Overview on the fault time instants.

Figure 5.29, instead, shows the current behaviour for all the files characterised by a fault (there is no distinction regarding the spot). It is clear that the presence of a fault does not imply some specific behaviour for the time-series, then this is a first example of the low influence that faults have on the production.

5.5.2 Energy-related analysis

After having verified that the presence of a fault does not influence too much the smooth operation of a welding process, it is interesting to evaluate if there are any issues related to the energy consumption. To this purpose, two main analysis are done:

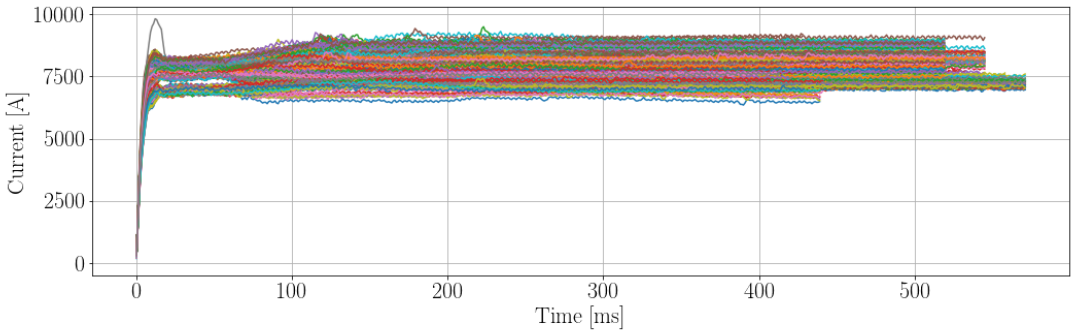


Figure 5.29: Current curves of processes affected by a fault.

- in figure 5.30 there are two examples of spots for which the percentage of welding process with faults is not extremely high (there are spots whose set of welding processes for the day under analysis feature a fault per file), neither extremely low or equal to 0. For them, an evaluation of the energy of each welding process is done, trying to understand if some particular behaviours can be detected. Looking at the two plots, it is clear that the energy is not related to the presence or the absence of a fault and this is another fact that demonstrates the low impact of faults on the production.
- In figure 5.31, instead, an analysis that is more related with the prediction of faults is performed. In fact, given a spot that as before is characterised by an average percentage of faults, the spot before and the one after it in the sequence are considered and their energies are evaluated, together with that of the spot "in the middle". The idea is to understand if the occurrence of a fault can be somehow revealed by particular behaviours before it or can influence in any way what happens after it. Looking at the figure, it can be said that, even if a fault occurs, no particular behaviour can be detected for the spot before and for that after the considered event.

It is then possible to conclude that the presence of faults is not related with the production quality and does not have an important impact. On the other hand, trying to predict it or to determine if it has an influence on the following events by looking at the behaviour of the welding process before and after the one under analysis seems difficult since there are no particular signals of the arrival of an anomaly.

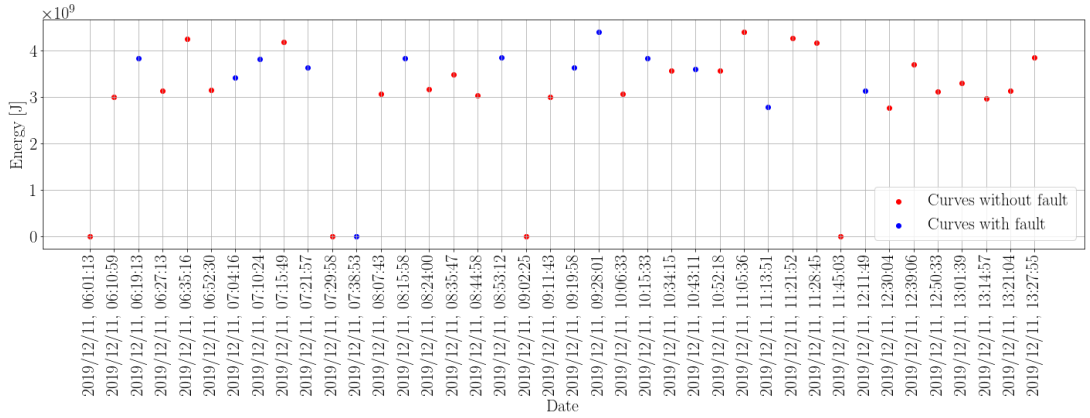


Figure 5.30: Example of energies analysis

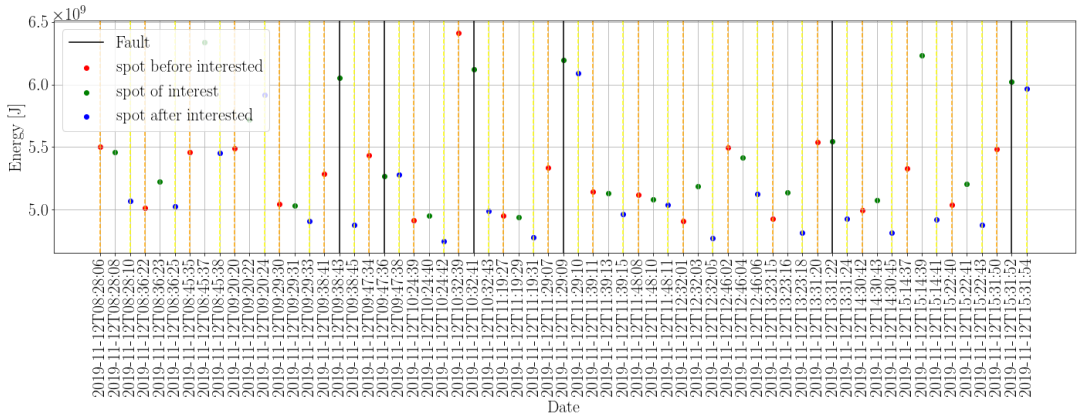


Figure 5.31: Energies analysis: comparison between a fault process and the ones before and after it.

Chapter 6

Conclusions

To conclude this work, there are a few things to say regarding what has been presented in the previous chapters.

First of all, it is clear that, even if there is not a pretty big difference between regular clustering with single points and time-series clustering for what concerns the base-concept and the algorithm steps, the latter requires a different approach and a different way of thinking for what concerns the computation of the distance measure. This is way the development of a custom-distance clustering algorithm has been necessary.

Talking about the algorithm itself, it has been demonstrated that it actually works on different sets of data (both synthetic and real) and it also provides satisfying results, proving its greater affinity with time-series, that are not only recognised but also divided into groups in an efficient and effective way. On the other hand, for what concerns the result, it has to be made clear that there is not a correlation between the clusters and groups of curves found in the various experiments and the provided metadata. An idea to fix this could be getting a better knowledge of the application domain by, for instance a more in-depth dialogue between technicians and domain experts and who performs the analysis on the provided data.

Talking about anomalies, it has been made clear that the presence of a fault can not be predicted by evaluating the energy consumption and also does not have any particular influence on what happens next. This is a important result (together with the outcomes of the energies analysis performed in the first part of section 5.5.2) and it allows to affirm that, despite the presence of anomalies, the quality of the production is not affected by them. Furthermore, even if there are welding scraps, there are no implications regarding energy consumption and this is also an added value because it confirms the low influence of the already described anomalies

on the production.

In the end, it can be said that in this particular context a periodic maintenance scheme can still work well and ensure good production quality. This statement finds further confirmation if we look at the analysis of anomalies, whose presence influences only minimally some parameters such as energy consumption and production quality. However, as described and demonstrated by the analysis of other works presented in section 1.3, the value of predictive analysis is undoubted and the transition to predictive maintenance schemes will become increasingly necessary, also thanks to the increasing influence and diffusion of the main aspects of industry 4.0.

Bibliography

[1] Wikipedia contributors. *Predictive maintenance — Wikipedia, The Free Encyclopedia*. `https://en.wikipedia.org/w/index.php?title=Predictive_maintenance&oldid=955920133`. [Online; accessed 3-June-2020]. 2020 (cit. on p. 1).

[2] Wikipedia contributors. *Data mining — Wikipedia, The Free Encyclopedia*. [Online; accessed 5-June-2020]. 2020. URL: `https://en.wikipedia.org/w/index.php?title=Data_mining&oldid=955848212` (cit. on p. 2).

[3] Christian Krupitzer, Tim Wagenhals, Marwin Züfle, Veronika Lesch, Dominik Schäfer, Amin Mozaffarin, Janick Edinger, Christian Becker, and Samuel Kounev. «A Survey on Predictive Maintenance for Industry 4.0». In: (2020) (cit. on p. 3).

[4] Kahiomba Kiangala and Zenghui Wang. «Initiating predictive maintenance for a conveyor motor in a bottling plant using industry 4.0 concepts». eng. In: *The International Journal of Advanced Manufacturing Technology* 97.9 (2018), pp. 3251–3271. ISSN: 0268-3768 (cit. on p. 3).

[5] Emiliano Traini, Giulia Bruno, Gianluca D'antonio, and Franco Lombardi. «Machine Learning Framework for Predictive Maintenance in Milling». eng. In: *IFAC PapersOnLine* 52.13 (2019), pp. 177–182. ISSN: 2405-8963 (cit. on p. 4).

[6] Leong Chee Him, Yu Yang Poh, and Lee Wah Pheng. «IoT-based Predictive Maintenance for Smart Manufacturing Systems». eng. In: *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)*. IEEE, 2019, pp. 1942–1944. ISBN: 9781728132488 (cit. on p. 4).

[7] Soonsung Hwang, Jongpil Jeong, and Youngbin Kang. «SVM-RBM based Predictive Maintenance Scheme for IoT-enabled Smart Factory». eng. In: *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*. IEEE, 2018, pp. 162–167. ISBN: 9781538652435 (cit. on p. 4).

[8] Elena Baralis and Paolo Garza. *Data Mining*. 2018 (cit. on p. 5).

[9] Elena Baralis and Tania Cerquitelli. *Clustering fundamentals*. 2010 (cit. on pp. 6–8).

[10] Wikipedia contributors. *Cluster analysis — Wikipedia, The Free Encyclopedia*. [Online; accessed 5-June-2020]. 2020. URL: `https://en.wikipedia.org/w/index.php?title=Cluster_analysis&oldid=960786594` (cit. on p. 6).

[11] F. Pedregosa et al. «Scikit-learn: Machine Learning in Python». In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830 (cit. on p. 31).