Università degli Studi di Padova

Department of Information Engineering

Master Thesis in Automation Engineering

Machine Learning approaches for Smart Monitoring of Sintering Equipment

Supervisor
Professor Gian Antonio Susto
Università degli Studi di Padova

Co-supervisor
Doctor Chiara Masiero
Statwolf
Doctor Federico Milan
Breton

Master Candidate
Samuele Mecenero

Academic Year

2021-2022

To my late grandfather, Tito. You were always present for me. I dedicate my work to you.

Sommario

Questa tesi nasce dalla collaborazione tra Statwolf, l'Università di Padova e Breton, azienda leader mondiale nella produzione di impianti per la lavorazione della pietra composita. In Breton, le lastre di pietra composita vengono compattate attraverso la vibrocompressione sotto vuoto. Questo progetto di tesi si propone di descrivere i passaggi necessari per implementare un approccio basato sul Machine Learning per monitorare un vibro-compattatore. Il vibrocompattatore ha come dotazione standard un gruppo di sei accelerometri che forniscono una serie temporale multivariata per ogni lastra lavorata dal macchinario. Abbiamo adottato un approccio basato sulla progettazione di feature specifiche poiché per ogni lastra, attraverso il processo di estrazione delle features, abbiamo tradotto il contenuto informativo delle serie temporali in quantità scalari. Abbiamo validato il processo di estrazione delle features e abbiamo sviluppato un sistema di rilevamento delle anomalie, che è il primo passo verso la progettazione di una soluzione di monitoraggio intelligente completa. A causa della mancanza di un set attendibile etichettato di dati anomali, che descrivesse tutti i possibili tipi di comportamento anormale, abbiamo adottato un approccio di apprendimento non supervisionato. Nello specifico, ci siamo concentrati sull'algoritmo di Isolation Forest per le sue elevate prestazioni di rilevamento delle anomalie, la sua efficienza computazionale e la sua adozione diffusa. Per fornire spiegazioni sulle previsioni dell'algoritmo, abbiamo sfruttato metodi di interpretabilità. In particolare, abbiamo utilizzato Depth-based Isolation Forest Feature Importance e Accelerated Model-agnostic Explanations, due metodi di interpretabilità, per fornire spiegazioni a livello globale e locale, rispettivamente. Infine, abbiamo utilizzato l'interpretabilità locale fornita da AcME come strumento di analisi delle cause principali per aiutare gli utenti a comprendere perché l'algoritmo considerava una lastra normale o anormale. I confronti con gli esperti di dominio di Breton hanno suggerito che l'approccio proposto è promettente.

Abstract

This work has been carried out thanks to the collaboration between Statwolf, the University of Padua and Breton, a global leading company manufacturing engineered stone processing plants. In Breton, vibro-compression under vacuum is used to compact engineered stone slabs. This thesis project aims to describe the steps necessary to implement a Machine Learning-based approach to monitor a vibro-compression machine. Vibro-compression machines have as standard equipment a group of six accelerometers that provide a multivariate time-series for each slab processed by the machinery. We have adopted a feature-based approach since for each slab, through the feature extraction process, we have translated the informative content of the time series into scalar quantities. We validated the feature extraction process, and then we developed a reliable anomaly detection system, which is the first step towards the design of a complete smart monitoring solution. We adopted an unsupervised learning approach, motivated by the lack of a reliable labelled set of anomalous data instances to describe all possible types of abnormal behaviour. Precisely, we focused on the Isolation Forest algorithm for its high detection performance, its computational efficiency and its widespread adoption. To provide insights on the predictions the Isolation Forest makes, we exploited interpretability methods. In particular, we used Depth-based Isolation Forest Feature Importance and Accelerated Model-agnostic Explanations, two interpretability methods, to provide explanations at a global level and local level, respectively. Finally, we employed local interpretability provided by AcME as a Root Cause Analysis tool to help users to figure out why the algorithm deemed a sample as normal or abnormal. Comparisons with Breton domain experts have suggested that the proposed approach is promising.

Contents

Listing of figures

Listing of tables

Listing of acronyms

AcME	Accelerated Model-agnostic Explanations
AD	Anomaly Detection
AI	Artificial Intelligence
AM	Advanced Monitoring
AS	Anomaly Score
DB	Distance Based
DIFFI	Depth-based Isolation Forest Feature Importance
DTW	Dynamic Time Warping
FB	Feature Based
FDI	Fault Detection and Isolation
HCI	Human-Computer Interaction
MHM	Machinery Health Monitoring
ML	Machine Learning
NN	Nearest Neighbor classifier
PCA	Principal Component Analysis
RMS	Root Mean Square
XAD	eXplainable Anomaly Detection
XAI	eXplainable Artificial Intelligence

1
Introduction

This thesis comes from the collaboration between Statwolf, Breton and the University of Padua. Statwolf is a knowledge-intensive service provider specialized in the development of Machine Learning-powered Decision Support Systems. Breton is a global leading company manufacturing machines to process natural stone and metals, as well as engineered stone processing plants. It is in this last business that this collaboration is born. This thesis project addresses the problem of smart monitoring in the context of engineered stone processing.

1.1 Problem description

Natural stones have been used as decorative construction materials due to their aesthetic qualification, accessibility and strength. However, they are often regarded as hard but fragile, so the combination with other materials to adequately offer more desirable qualities has been addressed. This new composite material is quoted as engineered stone. Engineered stone is thus defined as a composite material made from stone particles packed together by polymer resin or cement to obtain mechanical properties closed or superior to natural stones [1].
Sintering is a thermal process of converting loose fine particles into a solid coherent mass by heat and/or pressure without fully melting the particles to the point of melting [2].
Breton is the dominant supplier of equipment for making engineered stone. In Breton, vibro-compression under vacuum is used to compact engineered stone slabs. A schematic representation of a vibro-compression machine is depicted in figure 1.1.

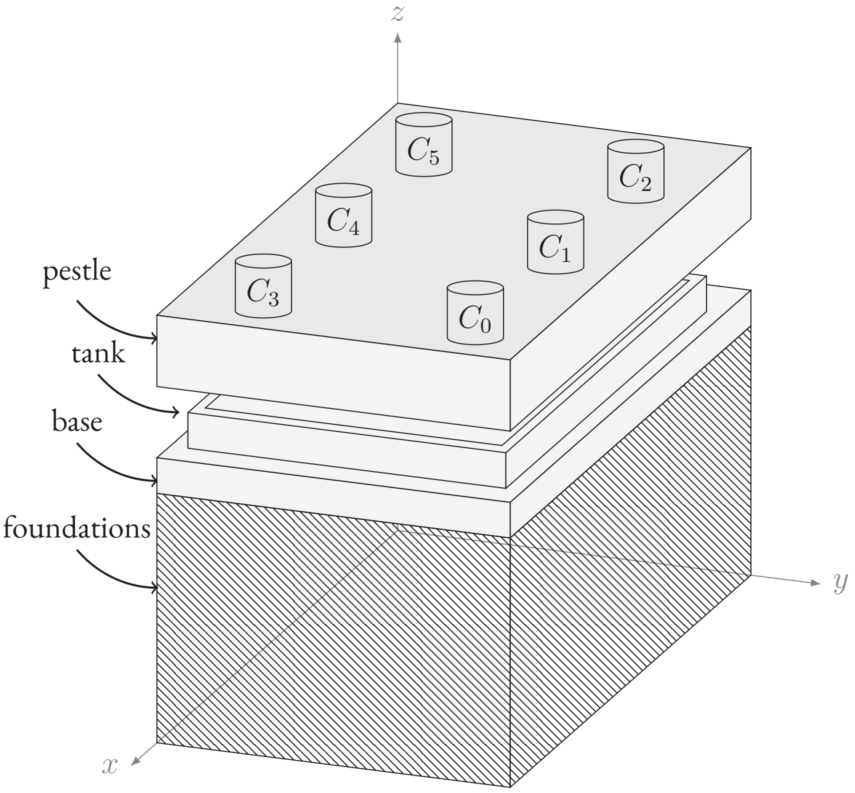


Figure 1.1: Vibro-compression machine.

In the operation of the vibro-compression machine, the pestle presses the stone particles contained in the tank to obtain an engineered stone slab. The incessant pressure can create some cracks in the base and can even provoke the detachment of the base from the foundations. Cracks represent a problem for Breton. For manufacturing companies, including Breton, the costs arising from any replacement or repair of equipment heavily affect the total cost even after the sale, both for the company and for the same customer. Thus, being able to detect and establish the cause of an anomaly from the first moment it manifests itself becomes critical to providing the right customer service and evaluating the type of assistance. From this criticality comes the main objective of this thesis project: try to establish the health condition of a vibro-compression machine, i.e. a healthy machine or a machine with cracks, through the study of vibration levels with Machine Learning (ML) algorithms. This work aims to describe the steps necessary to implement a ML based approach to monitor the health of a vibro-compression machine. Machinery health monitoring (MHM) embodies the ability to understand and react to the changing health of machinery and is well established within manufacturing environments

[3] [4]. A McKinsey study estimated that the appropriate use of MHM techniques by process manufacturers "typically reduces machine downtime by 30 to 50 percent and increases machine life by 20 to 40 percent" [5]. Vibro-compression machines have as standard equipment a group of six accelerometers arranged in a symmetrical fashion on the surface of the pestle. Figure 1.1 shows the arrangement of the accelerometers, they are denoted by C_0, C_1, C_2, C_3, C_4 and C_5. These accelerometers are used to monitor vibration levels. For Breton and its customers, vibration levels provide insights on the distribution of the stone particles contained in the tank. Exploiting vibration levels to also understand the health of the machinery would have a major impact, because it would allow Breton to provide customers with an additional service at no additional cost. Each accelerometer provides a continuous raw data stream $v_i(t)$ with $t \in \mathbb{R}, i \in \{0, \dots, 5\}$, which is turned into a finite-duration signal, $z_i(t)$, through sampling and windowing. Throughout this thesis, we consider the time-series $z_i(t)$ as a finite-length sequence of n ordered real values at time instants $t_{i,1}, \dots, t_{i,n}$. The time series $z_i(t)$ with $i \in \{0, \dots, 5\}$, provided by the accelerometers, can be exploited to define the multivariate signal $\{\mathbf{z}(\mathbf{t}) : t \in \mathbb{I}\}$, where:

$$\mathbf{z}(\mathbf{t}) \doteq \begin{bmatrix} z_0(t) \\ z_1(t) \\ z_2(t) \\ z_3(t) \\ z_4(t) \\ z_5(t) \end{bmatrix} \tag{1.1}$$

and $\mathbb{I}$ is the domain of the signal. The goal is to determine whether a multivariate signal $\mathbf{z}(\mathbf{t})$ has been collected from a healthy machine or a machine with cracks. The objective can be formulated either as a supervised learning problem or as an unsupervised learning problem. In supervised learning, the algorithm learns from the data that is provided, along with labels associated with the data (in the context of MHM, the label may be either healthy or failed). In unsupervised learning, the algorithm learns from the data, but no labels are provided. Thus, the unsupervised learner has to uncover useful information in the data without guidance from the labels.

1.2 Machine Learning approaches for time series

It is used the term classification to indicate the following supervised ML task: given a signal $\mathbf{x}$ belonging to some domain $\mathbb{X}$ as an input and a finite set $\mathbb{Y}$ of different classes (the output), the problem of classification amounts to finding a rule that associates $\mathbf{x}$ to one $\mathbf{y} \in \mathbb{Y}$ [6]. In the context of manufacturing, we often face classification problems when dealing with Fault Detection and Isolation (FDI) and Advanced Monitoring (AM). Just to provide an example, in the case of FDI, one output class can be related to the normal behavior of the vibro-compression machine, additional classes can be referred to known problems like machine with cracks or resonant machine, the input signals are time-series: the task of a FDI algorithm is to interpret signals coming from accelerometers to discern and understand the state of the overall system [7]. Time-series classification techniques can be essentially divided into two main branches:

- *Feature-based* (FB): FB methods perform a feature extraction procedure before the classification phase. A set $\mathbf{x}$ of p features is calculated over the time series. The idea underlying these methods is to capture signal statistics that identify a certain class of signals [8]. In theory, if a Gaussian process is weakly stationary then a second-order statistic is sufficient to characterize that signal; however, signals obtained from real-world scenarios are not stationary due to several nuisance factors and many more features may be necessary to summarize the informative content. When dealing with the learning phase in FB methods, the learning rule is based on the definition of a dataset of N observations and of a design matrix as:

$$D = \begin{bmatrix} \mathbf{x^{(1)}} & y^{(1)} \\ \mathbf{x^{(2)}} & y^{(2)} \\ \vdots & \vdots \\ \mathbf{x^{(N)}} & y^{(N)} \end{bmatrix} \in \mathbb{R}^{N \times (p+1)} \tag{1.2}$$

- *Distance-based* (DB): DB methods avoid the feature extraction phase in favor of the definition of suitable distances, among which the most common is dynamic time warping (DTW) [9]. Then, the classification phase is carried out through metric classifiers: one simple and often very effective choice is 1-Nearest Neighbor classifier (1-NN) [10].

The labels associated with a data instance denote whether that instance is normal or anomalous. It should be noted that obtaining labeled data that is accurate as well as representative of all types of behaviors, is often prohibitively expensive. Labeling is often done manually by a human expert and hence substantial effort is required to obtain the labeled training data set. Typically, getting a labeled set of anomalous data instances that covers all possible types of anomalous behavior is more difficult than getting labels for normal behavior. Moreover, the anomalous behavior is often dynamic in nature, for example, new types of anomalies might arise, for which there is no labeled training data [11]. Techniques that operate in unsupervised mode do not require training data, and thus are most widely applicable. The techniques in this category make the implicit assumption that normal instances are far more frequent than anomalies in the test data. If this assumption is not true, then such techniques suffer from high false negative alarm rate. Unsupervised anomaly detection (AD) [12] have emerged in recent years in the context of smart monitoring of complex systems. In the literature, unsupervised AD tools adopt two approaches:

- *multivariate approaches* based on tabular data: these approaches have the advantage of capturing multivariate anomalous behaviour that typically goes undetected by classic chart-based monitoring tools. When applied to time-series data, they entail the use of feature extraction procedures [13].
- *univariate approaches* working with time-series: these approaches typically work by predicting residuals, i.e., comparing measured and forecast time-series data, and raising an alarm as their difference exceeds a threshold [14].

Deep learning techniques are available for both the approaches, they typically need to be adapted to cope with discrete production data, where time-series are usually split into batches representing machine cycles [15]. This thesis project addresses the problem of smart monitoring of sintering equipment through an unsupervised learning approach. In particular, given that six accelerometers are placed in the vibro-compression machine, a *multivariate approach* is adopted.

1.3 Outline of the thesis

This section outlines the general structure of the thesis. Chapter 2 presents the operations necessary for the creation of the dataset. It presents an overview of the data pre-processing strategy and of the feature extraction phase. Furthermore, in the same chapter, we attempt to validate the feature extraction process. In Chapter 3 the methodology of the work is described. Chapter 4 reviews the results of several experiments. Chapter 5 sums up the potentialities and limitations of the current work and attempts to give some guidelines for future improvements.

2
Dataset

The data described in this chapter has been provided us by Breton. Specifically, the available data for the unsupervised AD task consists of a collection of 2 datasets for a total of $N = 712$ multivariate signals. The first dataset consists of $N_1 = 386$ multivariate signals that correspond to the processing of N_1 slabs on a healthy vibro-compression machine. The second dataset consists of $N_2 = 326$ multivariate signals that correspond to the processing of N_2 slabs on a failed vibro-compression machine. Each multivariate signal consists of 6 signals collected by the pestle's accelerometers. Raw signals have been turned into finite-duration digital signals through sampling and windowing. Finite-duration digital signals can be regarded as time series. Exclusively the use of end-to-end deep learning techniques allows machine learning models to be created without the need for feature engineering [16]. Unsupervised AD techniques adopted in this work are not straightforwardly applicable to time series. Albeit the time series were discrete time and of fixed length, the performance of the techniques would be poor if we considered each measurement sample as a feature, due to two main reasons: first, it is common to consider long sequences of $n > 10^4$ or even $n > 10^5$ samples; in these cases the space spanned by the time series is too large and sparse incurring in the "curse of dimensionality" problem. Second, considering the discrete values as independent features would be not reasonable, since they do not provide any information per se about the characteristics of the signal, since time-series values are strongly correlated on time and the feature extraction phase is exactly designed to highlight this correlation. Considering the dataset of N multivariate signals that describe the AD problem of interest, FB methods focus on finding a compact description $\mathbf{x} = [x_1, \ldots, x_p]$

of the multivariate signals $\mathbf{z}(\mathbf{t})$ such that $p \leq n$ (and typically $p \ll n$); all these N observations are collected into a matrix $X \in \mathbb{R}^{N \times p}$, called *feature matrix*, defined in equation 2.1.

$$X = \begin{bmatrix} \mathbf{x^{(1)}} \\ \mathbf{x^{(2)}} \\ \vdots \\ \mathbf{x^{(N)}} \end{bmatrix} \in \mathbb{R}^{N \times p} \tag{2.1}$$

In order to obtain the *feature matrix* X three operations were carried out:

1. Sampling and Windowing
2. Preprocessing
3. Feature extraction

In this chapter, these three important operations will be analyzed.

2.1 Sampling and Windowing

Signals are defined as functions of an independent variable and represent the evolution of physical entities that carry information. They can be classified on the basis of their domain $\mathcal{D}$ and codomain $\mathcal{C}$ [17]. The possible domains are:

- $\mathcal{D} = \mathbb{R}$: continuous time signals
- $\mathcal{D} = \mathbb{I}$, with $\mathbb{I}$ countable, $\mathbb{I} = \{\dots, t_{-1}, t_0, t_1, \dots\}$: discrete time signals

The possible codomains are:

- $\mathcal{C} = \mathbb{R}$: continuous amplitude signals
- $\mathcal{C} = \mathbb{I}$, with $\mathbb{I}$ countable and typically finite, $\mathbb{I} = \{x_1, x_2, \dots, x_M\}$: discrete amplitude signals

By combining the possible domains and codomains we obtain the following four signal classes:

- $\mathcal{D} = \mathbb{R}, \mathcal{C} = \mathbb{R}$: analog signals
- $\mathcal{D} = \mathbb{R}, \mathcal{C} = \mathbb{I}$: quantized analog signals

- $\mathcal{D} = \mathbb{I}, \mathcal{C} = \mathbb{R}$: sampled signals
- $\mathcal{D} = \mathbb{I}, \mathcal{C} = \mathbb{I}$: digital signals

The only class of signals that computer systems can exactly represent and deal with is the class of digital signals. Breton provided us a set of $N = 712$ multivariate digital signals. Each multivariate signal consists of 6 finite-duration digital signals. These signals have domain $\mathcal{D} = \mathbb{I}$, where $\mathbb{I} = \mathbb{Z}(T_S)$, i.e. $t_k = kT_S$. The sampling frequency of the signals were $F_S = 1654$Hz, accordingly the sampling period of the signals were $T_S = {F_S}^{-1}$. Each signal was collected manually by a Breton expert that started the collection with a start command and ended it with a stop command. The manual start and stop did not allow to have time-series with the same length. This is a problem because it is more difficult to compare time series with different lengths. Table 2.1 shows descriptive statistics of time series. It is possible to notice that the standard deviation is high, this indicates that the lengths of the time series are distributed over a wide range.

Number of time series	712
Average length	65.7s
Standard deviation	13.2s
Minimum length	39.5s
First quartile	56.4s
Median	66.4s
Third quartile	67.4s
Maximum length	97.4s

Table 2.1: Time series' descriptive statistics.

A possible solution to this problem could be to have an automatic start and stop. A simple idea to implement could be to start the collection when vibration levels exceed a threshold δ and stop it after a time interval τ_{stop}. δ must be chosen by domain experts, who, having a thorough knowledge of the machinery, would be able to choose the value that represents the beginning of processing. As far as τ_{stop} is concerned, there are two possible choices: choose the maximum duration of the processing of a slab or rely on domain experts who would be able to suggest the most suitable value.

2.2 Preprocessing

After the sampling and windowing phase, where raw signals have been transformed into finite-duration digital signals, the preprocessing phase occurred. The goal of this phase was to design a sample rate conversion system to convert from the original sampling frequency F_S to another sampling frequency F'_S. The original sampling frequency $F_S = 1654\text{Hz}$ was quite high, accordingly the number of samples per signal was high too. Reducing the number of samples per signal would have meant a reduction in computational cost. Our objective was to have a reduction in computational costs as long as we limited the loss of information due to down-sampling. A decimator is a system that performs a down-sampling by a factor M of the input signal by preserving only one every M input samples. Figure 2.1 shows a block diagram of a decimator.

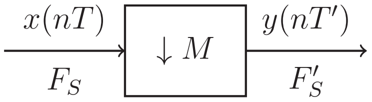


Figure 2.1: Block diagram of a decimator.

This system performs a time-domain transformation, its input-output equation is:

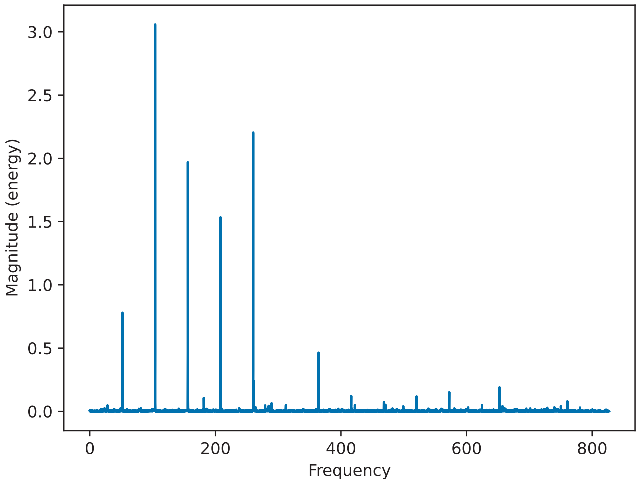
$$y(nT') = x(nT') = x(nMT) \tag{2.2}$$

The input-output equation of the decimator in terms of the Fourier transform is given by:

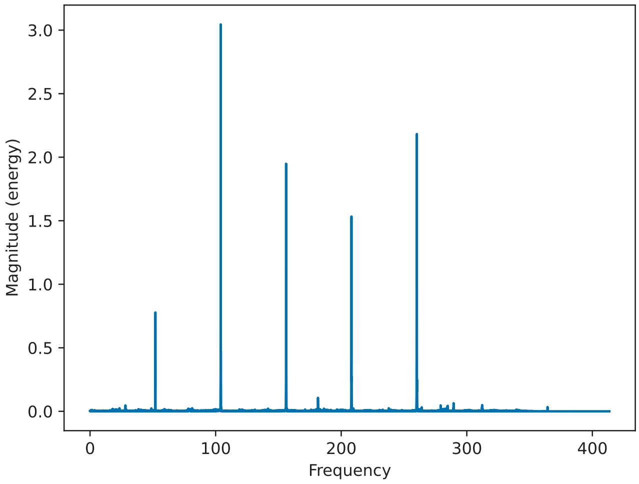
$$\widetilde{Y}(f) = \sum_{k=0}^{M-1} \widetilde{X}(f - kF'_S) \tag{2.3}$$

Notice that $\widetilde{Y}(f)$ is a periodic function with period $F'_S = \frac{1}{M}F_S$, which is obtained as the superimposition of M shifted versions of the Fourier transform $\widetilde{X}(f)$, that is periodic with period F_S. In general, the decimation from $x(nT)$ to $y(nT')$ causes a loss of information which can be explained by the overlapping, in the frequency domain, of the terms in the summation 2.3. In case $M = 2$, if the input signal $x(nT)$ is bandlimited to the interval $\left[\frac{-F'_S}{2}, \frac{F'_S}{2}\right]$ there is no overlapping of the shifted versions of the Fourier transform $\widetilde{X}(f)$, and therefore the information contained in input signal can be entirely preserved. The signals available were clearly not bandlimited, but most of their frequency content was contained in the interval $\left[\frac{-F'_S}{2}, \frac{F'_S}{2}\right]$.

We got bandlimited signals by applying a low-pass filter with a cutoff frequency of $\frac{F'_S}{2}$. With such a filter, the Fourier transform of the signal $y(nT')$ at the output of the decimator is equal to the Fourier transform of the input signal $x(nT)$ in the frequency interval $\left[\frac{-F'_S}{2}, \frac{F'_S}{2}\right]$. The unavoidable loss of information is limited because most of the frequency content of the signals is contained in the interval $\left[\frac{-F'_S}{2}, \frac{F'_S}{2}\right]$ as depicted in figure 2.2.



(a) Magnitude spectrum of a signal before down-sampling. It is possible to notice that most of the frequency content of the signal is contained in the interval $\left[0, \frac{F'_S}{2}\right]$, where $F'_S = 827\text{Hz}$.



(b) Magnitude spectrum of a signal after down-sampling.

Figure 2.2: Magnitude spectrum of a signal before and after down-sampling.

2.3 Feature extraction

After applying the decimation to each signal, we moved on to the feature extraction phase. The feature extraction phase consists of translating the informative content of time-series data into scalar quantities. This phase may be a time-consuming step that requires the involvement of process experts to avoid loss of information in the making. Hence, to design proper features, we have exploited the information provided to us by the Breton experts. Specifically, two information have been employed:

- In the processing of a slab, a vibro-compression machine go through three different phases. In the first phase, vibration levels of the machinery increase until a operating value is reached. In the second phase, vibration levels are maintained at the operating value. In the third phase, vibration levels decrease to zero.

- When a vibro-compression machine is failed, the six time series related to the processing of a slab have a heterogeneous behaviour.

The first information led us to decide that each time series should be divided into segments. The optimal split would have been into three segments, where each segment represented a phase of the operation of the machinery. However, we did not have enough information to implement this optimal split. Therefore, we opted for another split. We decided to divide each time series into four segments of equal length. In this way we knew that the first segment would capture a part, if not the whole, first phase, the second and third segment would capture a considerable part of the second phase, and the third segment would capture the third phase, which we knew was the shortest. Figure 2.3 depicts the split in four segments of a multivariate time series related to an engineered stone slab. In the figure, a slab processed by a healthy machinery is considered.

The second information led us to understand that we had to consider each of the six time series related to the processing of a slab. Figure 2.4 shows that when a vibro-compression machine is failed, the six time series related to the processing of a slab have a heterogeneous behaviour.

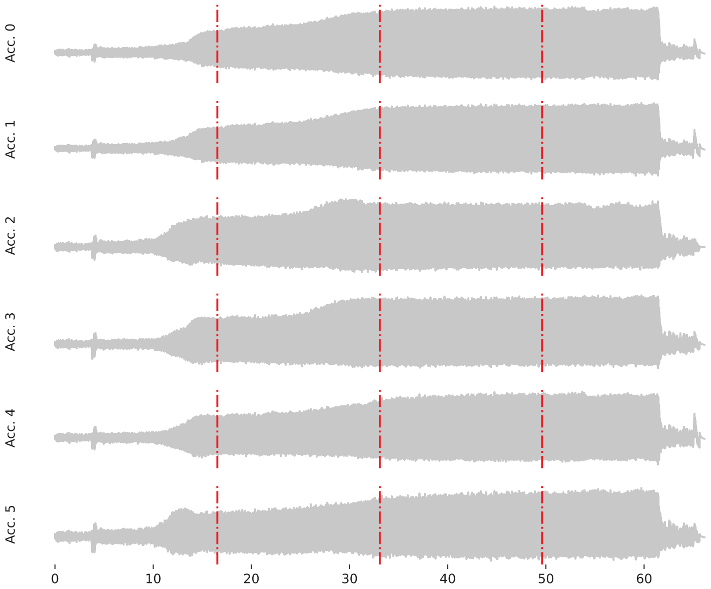


Figure 2.3: Multivariate time series related to a slab processed by a healthy vibro-compression machine.

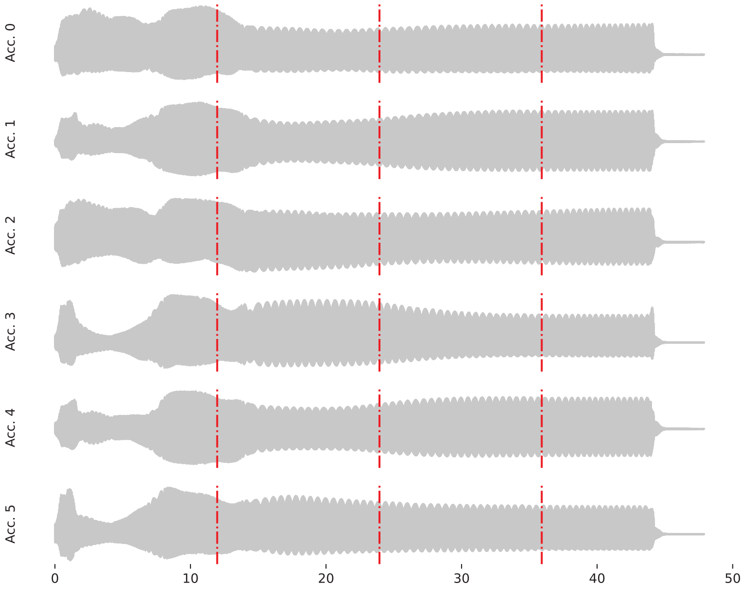


Figure 2.4: Multivariate time series related to a slab processed by a failed vibro-compression machine.

These two information have allowed us to extract features properly. For each of the N multivariate time-series a set of $p = 40$ features have been extracted. 36 features have been extracted from the average time series $\bar{z}(\cdot)$, defined as:

$$\bar{z}(kT_S) \doteq \frac{1}{6}\sum_{i=0}^{5} z_i(kT_S), \quad \forall k \in \mathbb{I} \tag{2.4}$$

where $z_i(kT_S)$ is the value, at the time instant kT_S, of the time series collected by the i-th accelerometer. The average time series have been split in four segments, for each segment 9 features have been extracted. There were both time-related and frequency-related features. The time-related features were:

- Mean
- Root Mean Square (RMS)
- Standard deviation
- Skewness
- Kurtosis
- First quartile
- Third quartile

The definitions of these statistics can be found in [18]. The frequency-related features were:

- Fundamental frequency
- Variance of the magnitude spectrum in a neighbourhood of the fundamental frequency

We decided to take these two frequency-related features because we observed that failed machines had a different behaviour in frequency with respect to healthy machines. Finally, 4 features have been extracted from the time series $z_i(t)$ with $i \in [0, \ldots, 5]$. Each time series has been split into four segments, and for each segment the sample mean has been computed. Then, the variance of the six sample means has been calculated, thus obtaining 4 scalar features. An instance is a vector that contains the features extracted from a multivariate signal. An instance is a row in the *feature matrix X*. We got a *feature matrix X* that has N rows corresponding to the N multivariate signals and p columns corresponding to the set of p features extracted for each signal.

2.4 Feature extraction validation

Machine learning is a powerful tool for gleaning knowledge from massive amounts of data. While a great deal of machine learning research has focused on improving the accuracy and efficiency of training and inference algorithms, there is less attention on the equally important problem of monitoring the quality of data fed to machine learning [19]. The importance of this problem is hard to dispute: poor input data can nullify any benefits on speed and accuracy for training and inference. This argument points to a data-centric approach to machine learning that treats training and serving data as an important production asset, on par with the algorithms and infrastructure used for learning. The process of extracting relevant features from the data to train ML algorithms is called feature engineering. Feature engineering is vital to data science as it produces reliable and accurate data and algorithms are only as good as the data fed to them [20]. We have exploited some tests made on vibro-compression machine by Breton to validate the feature extraction process. Breton tested diverse setups of the machinery on a rubber slab specifically made for testing. The setups tested were 5, some tests have been repeated several times, some other no. Table 2.2 describes the tests carried out.

Setup tested	Number of tests carried out	Name of the tests
Setup 1	2	$Test\_1A, Test\_1B$
Setup 2	3	$Test\_2A, Test\_2B, Test\_2C$
Setup 3	1	$Test\_3$
Setup 4	1	$Test\_4$
Setup 5	1	$Test\_5$

Table 2.2: Vibro-compression machine tests.

For each test, a multivariate time-series has been provided by the accelerometers arranged on the pestle. The tests had different duration, to make it the same a rectangular window was applied. Lastly, a vector $\mathbf{x}^{(i)}$ of features has been extracted for each test. Given that the duration of the time-series was equal to 5 seconds, it has been decided not to divide the signals into 4 segments because they would have been too short. Specifically, the feature extraction procedure described in section 2.3 has been applied considering the whole time-series as a unique segment. Therefore, a set of $p' = 10$ features have been extracted from each test. The features extracted were:

- Standard deviation referred as *std*

- Skewness referred as *skew*
- Kurtosis referred as *kurt*
- Root Mean Square referred as *RMS*
- Sample mean referred as *mean*
- First quartile referred as *q25*
- Third quartile referred as *q75*
- Fundamental frequency referred as f_0
- Variance of the magnitude spectrum in a neighbourhood of the fundamental frequency referred as *Var(f_0)*
- Variance of the 6 sample means referred as *Var(mean)*

Since a vector $\mathbf{x}^{(\mathbf{i})}$ of features is a row in the *feature matrix* X', we got a *feature matrix* X' that has $N' = 8$ rows corresponding to the N' tests and $p' = 10$ columns corresponding to the set of p' features extracted for each test. To validate the process of feature extraction, we wanted to show that feature vectors related to different setups of the machinery are very different, while feature vectors related to the same setup of the machinery are very similar. Indeed, features are considered relevant if they are able to identify different behaviors of the vibro-compression machine. To show that we have chosen relevant features, we have employed Principal Component Analysis (PCA). The goal of PCA is to find the sequence of orthogonal components that most efficiently explains the variance of the observations. The advantage of PCA is that it finds a lower-dimensional representation, while preserving the maximum amount of information from the original variables. For the feature matrix X', PCA yields an orthogonal decomposition of X' that is optimal for a given number of principal components. The principal component decomposition provides the minimum mean squared error approximation to X'. Moreover, the explained variation of the excluded principal components converges to zero as K increases, where K denotes the retained number of principal components [21]. We have exploited the PCA to obtain a 3-dimensional representation of the feature vectors. Figure 2.5 depicts the variance explained by each of the principal components. It is possible to note that the cumulative explained variance for the three components is very close to 100%. In figure 2.6 the weights for each original variable when calculating the principal components are reported.

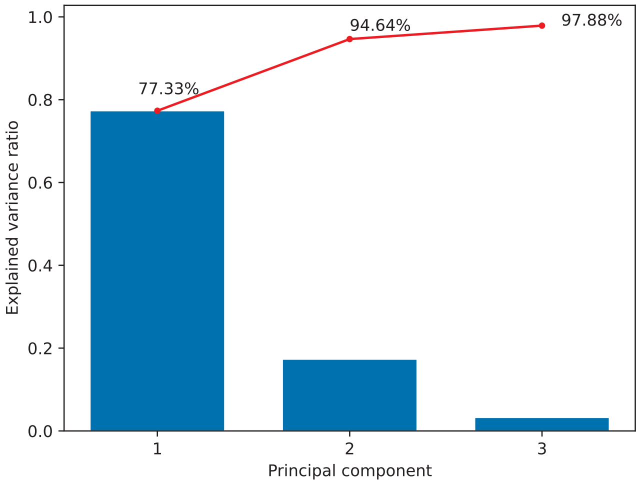


Figure 2.5: Cumulative and individual explained variance.

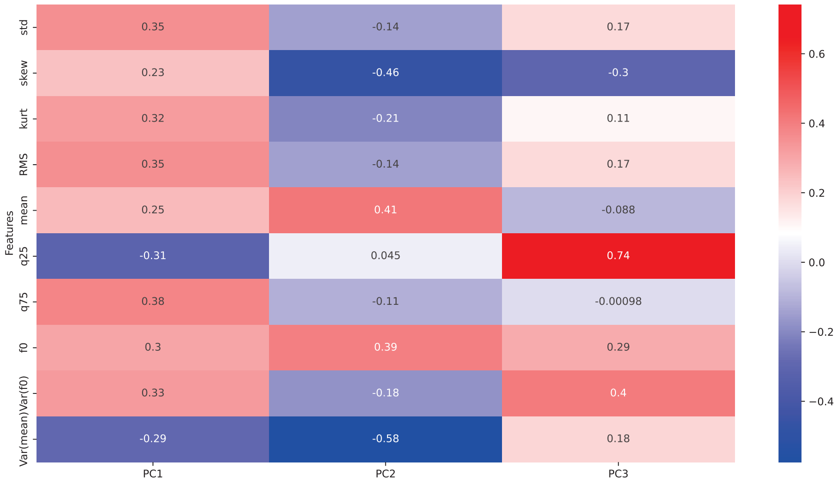


Figure 2.6: PCA - Weights for each original variable when calculating the principal components.

Figure 2.7 shows the first three principal components of these data. The 3-dimensional representation of feature vectors provided by PCA supports the feature engineering process. In fact, from the figure it is possible to notice two clusters composed of tests related to the same machinery setup. Specifically, the tests related to Setup 1 are very close, similarly, the tests related to Setup 2 are very close too. Quite the opposite, tests related to different setups are distant from each other.

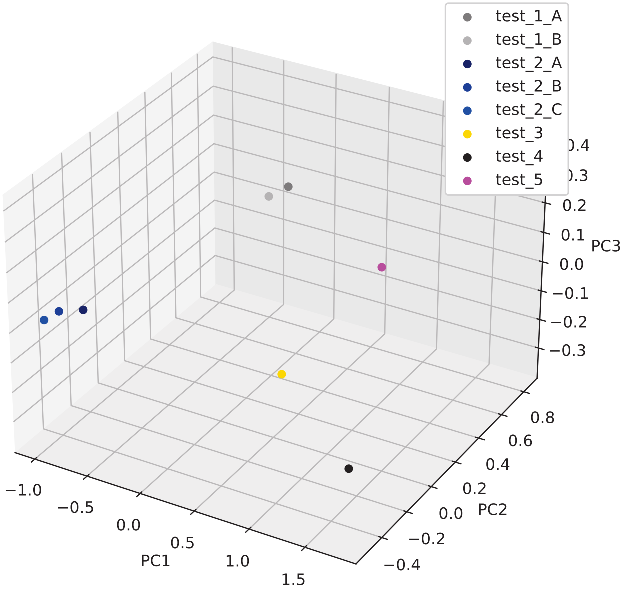
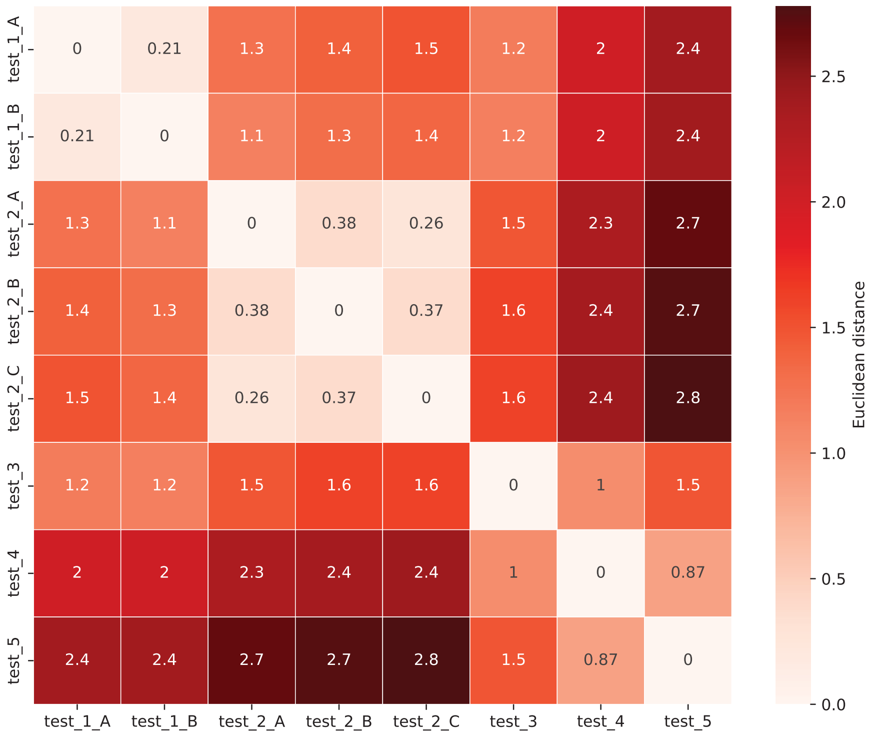


Figure 2.7: First three principal components of the data.

To further support the feature engineering process, a dissimilarity matrix, which uses as a metric distance the measure called Euclidean Distance, has been computed. The Euclidean Distance between two feature vectors $\mathbf{x^{(i)}}$ and $\mathbf{x^{(j)}}$ is defined by:

$$d_{ij} \doteq \left(\sum_{k=0}^{p'} \left| x^{(i)}(k) - x^{(j)}(k) \right|^2 \right)^{\frac{1}{2}} \tag{2.5}$$

The dissimilarity matrix is a $N' \times N'$ matrix $\mathbf{D}$, where N' is the number of tests, and each element d_{ij} records the dissimilarity between the ith and jth tests. A heatmap representing the dissimilarity matrix $\mathbf{D}$ is reported in figure 2.8.



	test_1_A	test_1_B	test_2_A	test_2_B	test_2_C	test_3	test_4	test_5
test_1_A	0	0.21	1.3	1.4	1.5	1.2	2	2.4
test_1_B	0.21	0	1.1	1.3	1.4	1.2	2	2.4
test_2_A	1.3	1.1	0	0.38	0.26	1.5	2.3	2.7
test_2_B	1.4	1.3	0.38	0	0.37	1.6	2.4	2.7
test_2_C	1.5	1.4	0.26	0.37	0	1.6	2.4	2.8
test_3	1.2	1.2	1.5	1.6	1.6	0	1	1.5
test_4	2	2	2.3	2.4	2.4	1	0	0.87
test_5	2.4	2.4	2.7	2.7	2.8	1.5	0.87	0

Figure 2.8: Heatmap representing the dissimilarity matrix.

The heatmap underlines the presence of two clusters. The first cluster is composed by $Test\_1A$ and $Test\_1B$, whereas the second cluster is composed by $Test\_2A$, $Test\_2B$ and $Test\_2C$. The tests that belong to the first cluster are both related to same machinery setup, the same thing applies to the second cluster. As observed in PCA, tests related to different setups of the machinery are considered to be very dissimilar. For example, tests related to Setup 2 and the test related to Setup 5 are very dissimilar. The first three principal components of the data and the heatmap representing the dissimilarity matrix have validated the feature engineering process. In particular, they have shown that the features extracted are able to identify different setups, and hence behaviors, of the vibro-compression machine.

3
Proposed approach

The growing complexity of industrial processes has generated a constant need for improvements in the safety and reliability of systems. These two properties can be achieved through preventive actions resulting from process monitoring, system reconfiguration, periodic maintenance, and the installation of safe components [22]. In Breton, the production process of engineered stone slabs is very complex. Vibro-compression under vacuum represents only one of the phases that make up the process. The presence of faults in vibro-compression machine may result in degradation of product quality, increased operating costs and production stoppages. This is why smart monitoring of the machinery is of paramount importance. Smart monitoring can be based on knowledge of the exact model or a model with considerable approximation of the system. Enabling a physical interpretation of the process and consequently a better understanding of the behavior of the analyzed variables, represents an advantage in practical applications. However, obtaining these models becomes difficult or almost impossible when dealing with complex, large-scale systems, and with operations involving a significant number of variables. A data-driven approach appears as a possible solution because it is based on available information on the system's behavior and historical process data. Due to the complexity to obtain an exact physical interpretation of the system, we have adopted a data-driven approach for the smart monitoring of vibro-compression machine. The very first step towards the design of a complete smart monitoring solution is the development of a reliable Anomaly Detection system.

3.1 Anomaly Detection

Anomaly detection (AD) is the process of finding outliers in a given dataset [23]. Outliers are the data objects that stand out amongst other data objects and do not conform to the expected behavior in a dataset. Hence, an outlier is always defined in the context of other objects in the dataset. If the training dataset has objects with known anomalous outcomes, then any of the supervised data science algorithms can be used for anomaly detection. In addition to supervised algorithms, there are unsupervised algorithms whose whole purpose is to detect outliers without the use of a labeled training dataset. Motivated by the lack of a reliable labeled set of anomalous data instances that covers all possible types of anomalous behavior, an unsupervised learning approach has been adopted. AD methods typically define a so-called Anomaly Score (AS), a quantitative index associated with the degree of 'outlierness' of the observation under exam.

3.1.1 Isolation Forest

Isolation Forest (IF) is a popular AD algorithm [24] [25]. IF is an unsupervised AD algorithm leveraging an isolation procedure to infer a measure of outlierness, the anomaly score, for each data point: the isolation procedure is based on recursive partitioning and aims at defining a region in the data domain where only the data point under examination lies. The underlying mechanism of IF is based on the reasonable hypothesis that the isolation procedure for outliers requires a limited number of iterations, while the isolation of inliers generally needs a larger number of recursive partitions. The isolation procedure defines a tree-like model of decisions, called Isolation Tree. Each node in the tree is linked to a variable, and its children, if present, are determined based on a splitting value. Many Isolation Trees are computed in an IF selecting variables and values for splitting at random, making the IF an ensemble method. AS associated with an observation is computed in IF by evaluating the mean path length of such observation on the various Isolation Trees. In order to obtain the predicted binary labels a thresholding operation is performed on the AS associated to all observations. The choice of the threshold value is based on the expected fraction of outliers in the dataset. The IF model has three main advantages:

- high detection performance (often even with default hyperparameters values, with no tuning required)
- computational efficiency

- possibility to parallelize its computation (thanks to its ensemble structure)

Given these advantages and the widespread adoption of IF in AD tasks [26][27][28], we have decided to focus on such an algorithm in this work. The AS, provided by IF, can be exploited in an industrial environment as a sort of health factor of the monitored system and monitoring policies based on thresholds on the AS can be in place: when the AS overcomes a certain threshold, inspections or other monitoring actions can be triggered. However, the main issue in relying on such an approach is that Root Cause Analysis is non-trivial. AD methodologies do not provide a simple way to indicate which are the causes of variations in AS, leaving the human operator without guidance on how to react after a threshold crossing has happened. This is a typical case where the need of interpretability is of paramount importance to make decisions based on a ML module. A definition of interpretability is: Interpretability is the degree to which a human can understand the cause of a decision [29]. The higher the interpretability of a machine learning model, the easier it is for someone to comprehend why certain decisions or predictions have been made [30]. To address this need, Explainable Artificial Intelligence (XAI) proposes to make a shift towards more transparent Artificial Intelligence (AI) [31]. It aims to create a suite of techniques that produce more explainable models whilst maintaining high performance levels.

3.2 Explainable Artificial Intelligence

Recently, the notion of Explainable Artificial Intelligence has seen a resurgence, after having slowed since the burst of work on explanation in expert systems over three decades ago. This resurgence is driven by evidence that many AI applications have limited take-up, or are not appropriated at all, due to ethical concerns [32] [33] and a lack of trust on behalf of their users [34] [35]. The running hypothesis is that by building more transparent, interpretable, or explainable systems, users will be better equipped to understand and therefore trust the intelligent agents. It is important to note that the solution to explainable AI is not just 'more AI'. Ultimately, it is a human-agent interaction problem. Human-agent interaction can be defined as the intersection of artificial intelligence, social science, and human-computer interaction (HCI). XAI is just one problem inside human-agent interaction; see figure 3.1.

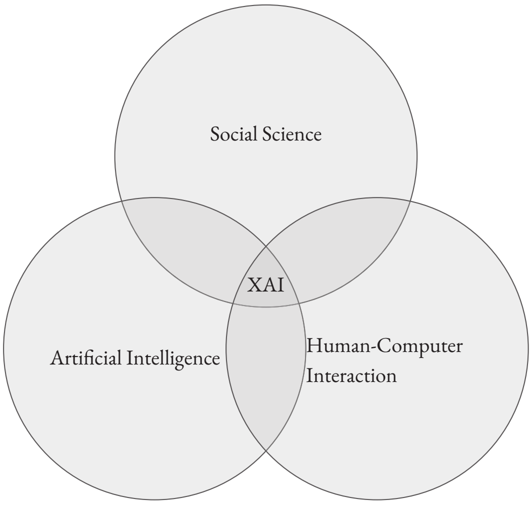


Figure 3.1: Scope of Explainable Artificial Intelligence.

To avoid limiting the effectiveness of the current generation of AI systems [36], XAI proposes creating a suite of ML techniques that:

- produce more explainable models while maintaining a high level of learning performance (e.g., prediction accuracy)
- enable humans to understand, appropriately trust, and effectively manage the emerging generation of artificially intelligent partners

When we talk about an explanation for a decision, we generally mean the need for reasons or justifications for that particular outcome, rather than a description of the inner workings or the logic of reasoning behind the decision making process in general. Using XAI systems provides the required information to justify results, particularly when unexpected decisions are made. XAI for operations in industry should help the end-user of ML to understand the model and draw conclusions from the prediction [37]. IF does not provide insights on the predictions it makes. Hence, in order to help end-users, i.e. Breton technicians, to understand and draw conclusions from the predictions, we should explain the algorithm predictions. In particular, we should design an Explainable Anomaly Detection (XAD) system.

3.2.1 DIFFI

The AS, provided by IF, can only rank the objects on how much the algorithm believes the objects to be outliers. Accordingly, users can find it hard to understand the decisions made by the model. From this comes the need to yield insights on the AS provided by IF to the end-users. To meet this need, ML interpretability methods can be exploited. ML interpretability methods can be classified into model-specific and model-agnostic ones. The former, also known as ad hoc methods, are designed for a specific class of models. The latter, also known as post hoc methods, can be applied on top of every ML model. Depth-based Isolation Forest Feature Importance (DIFFI) is a model-specific method addressing the need for interpretability for the IF detector [38]. By providing a quantitative measure of feature importance in the context of the AD task, DIFFI allows to describe the behavior of IF at global and local scale, providing insightful information that can be exploited by final users of an IF-based AD solution to get a better understanding of the underlying process and to enable root cause analysis. Global interpretability methods aim to provide explanations of the model as a whole, local interpretability methods aim to provide explanations associated with individual predictions. DIFFI method has two important advantages:

- It is a completely unsupervised approach
- It does not require any modification in the original IF procedure, but the access to the structure of resulting Isolation Trees

An unsupervised feature importance evaluation method is pivotal in this scenario. Specifically, if the estimated feature importance scores aligned well with human prior knowledge, users would be more prone to lessen the supervision and safely give more autonomy to the machine. The need for interpretability to foster trust and enable Root Cause Analysis besides significantly smaller computational costs with respect to the current state-of-the-art method SHAP [39] were the reasons that prompted us to use DIFFI.

3.2.2 AcME

Accelerated Model-agnostic Explanations (AcME) is an interpretability approach that quickly provides feature importance scores both at the global and the local level [40]. It is a model-agnostic interpretability approach designed with execution speed in mind. This requirement is of paramount importance in human-in-the-loop applications where users need to take corrective actions quickly. The importance scores provided by AcME rely on perturbations of the data based on quantiles of the empirical distribution of each feature. These perturbations are performed with respect to a reference point in the input space. Experimental results suggest that AcME produces global explanations similar to those provided by SHAP, in a fraction of the computation time. As for local interpretability, differently from SHAP, AcME provides a simple what-if tool that allows users to figure out how changes in feature values may affect the predictions. AcME is suitable for unsupervised Anomaly Detection, where observations are assigned an anomaly score to detect the most abnormal behaviours. Exploiting the local interpretability, we could focus on a sample and use AcME to evaluate how changes in the input feature values would impact the corresponding anomaly score. Thus, AcME would act as a Root Cause Analysis tool, in the sense that it may help users to figure out why the algorithm deemed a sample as normal or abnormal. In the scenario of Anomaly Detection, where prompt corrective actions can translate into a reduced waste of money, time, and materials, the execution speed of interpretability procedures is even more relevant.

4 Experimental results and evaluations

This chapter expands on the experimental phase of the project, providing an account of its design and a thorough discussion of the results and their implications. The first step of this phase is data exploration, an essential step to understand the structure of the data and the relationships within the dataset. As the second step of the experimental phase, we have moved into the AD task. In this step we focused on the IF algorithm, and in particular on its interpretability to get a better understanding of the underlying process and to enable root cause analysis. Finally, we have adopted Root Cause Analysis to determine the root causes of two anomalous instances in order to implement the appropriate improvement action.

4.1 Data exploration

Before venturing into any advanced analysis of data using statistical, machine learning, and algorithmic techniques, it is essential to perform preliminary data exploration to study the basic characteristics of a dataset. Data exploration helps with understanding data better, to prepare the data in a way that makes advanced analysis possible, and sometimes to get the necessary insights from the data faster than using advanced analytical techniques. Visualizing data is one of the most important techniques of data exploration. A visual plot of data points provides an instant grasp of all data points condensed into one chart. To get a visual plot of the $N = 712$ data points we have exploited PCA. In particular, we have employed PCA to get a 3-dimensional representation of the data points. Figure 4.1 depicts the variance explained by each of the principal components. It is possible to note that the cumulative explained variance for the three components is close to 80%. In figure 4.2 the weights for each original variable

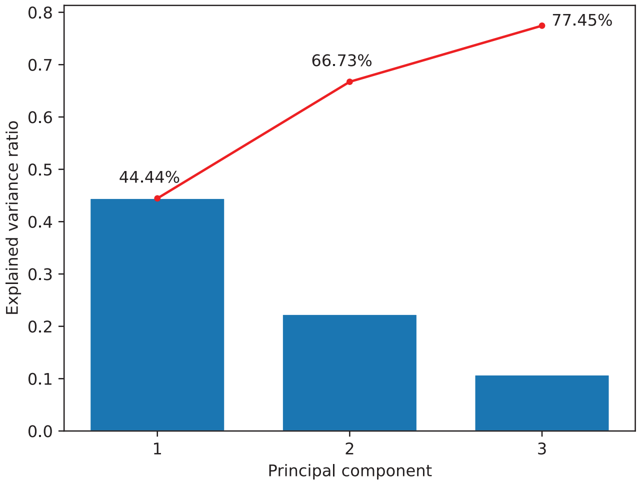


Figure 4.1: Cumulative and individual explained variance.

when calculating the principal components are reported. We recall that we had a collection of N data points that correspond to the processing of N slabs by a vibro-compression machine. $N_1 = 386$ data points were related to a healthy machinery whereas $N_2 = 316$ data points were related to a failed machinery. Of each slab we knew:

- weight
- thickness
- material composition

Figures 4.3 and 4.4 depict the three principal components of the data points, where the third information listed above has been used in the figures to highlight the 7 different material compositions. We want to underline that the slabs related to *material code 4*, *material code 5*, *material code 6* and *material code 7* have been processed by a failed machinery. In fact, from figure 4.4 it is possible to note that data points related to a healthy machinery present a different behavior with respect to data points related to a failed machinery. Precisely, the data points related to a failed machinery are more spread. This visual plot of data points was extremely helpful because it allowed us to understand that the features extracted from each multivariate time series could

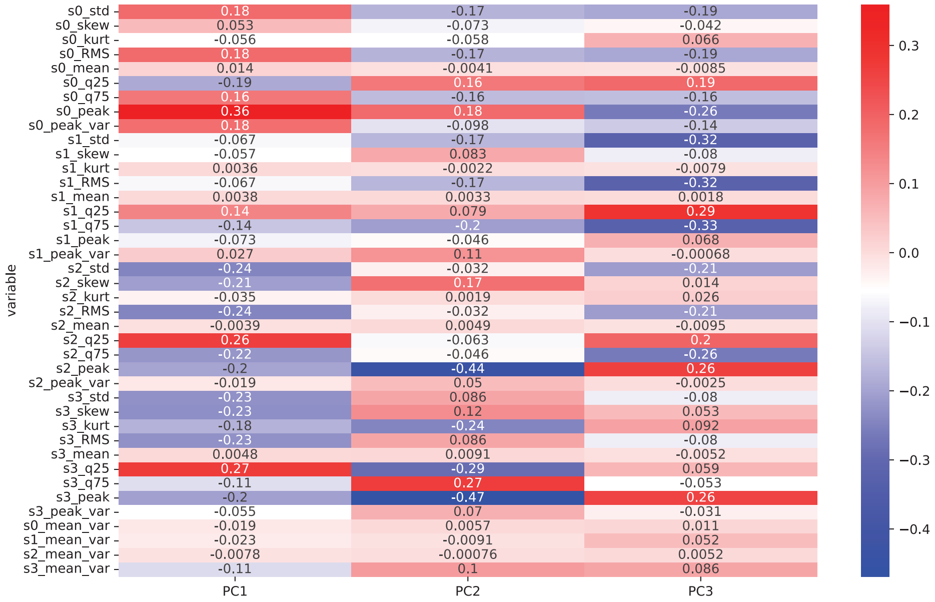


Figure 4.2: Weights for each original variable when calculating the principal components.

identify the slabs according to their composition. Besides, the plot suggested training a separate AD model for each different material composition. Indeed, the IF algorithm relies on the idea that abnormal samples are simpler to isolate than normal ones, which form dense clusters. Consider the case when we have data belonging to classes corresponding to different material compositions, where one has significantly fewer, sparser samples. If we had applied the IF algorithm to the whole dataset, most of the samples coming from the minority class (i.e., those coming from the failed machine) would have been deemed abnormal. These considerations led us to train specialized AD algorithms for each composition. Moreover, unsupervised AD assumes that most samples are normal, but this is not true for the available instances coming from the failed machine. Thus, we have decided to focus only on the instances related to the most present material composition, namely *material code 3*.

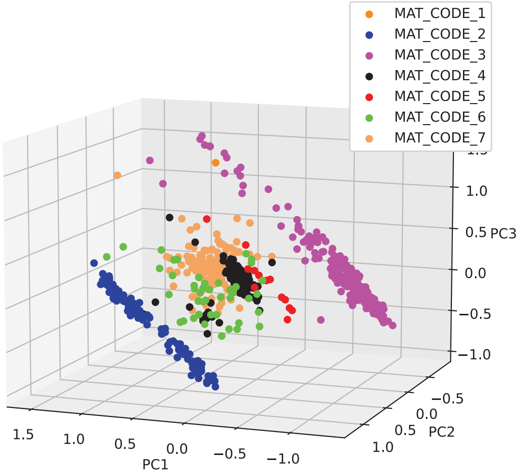


Figure 4.3: First three principal components of the data.

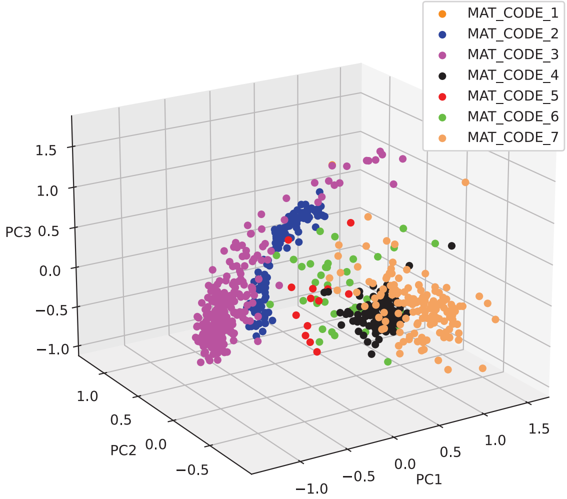


Figure 4.4: First three principal components of the data.

4.2 Unsupervised Anomaly Detection

Before proceeding with the AD task, we studied the instances related to *material code 3*. We employed the thickness and weight of each slab to study its distribution. Precisely, we made a scatter plot to display values for the thickness and weight; see figure 4.5. From the figure, a bimodal distribution of both the thickness and weight can be observed. These bimodal distributions lead to the formation of two clusters. This visual plot of data points was extremely helpful, because it allowed us to understand that we had to develop an AD system that did not depend on the weight and on the thickness of the slabs. If the AD system was weight dependent, some slabs could have been labeled as anomalous only because they had a mass very dissimilar from the average of the masses. From figure 4.5 it is even possible to note that there is a data point that has a thickness value of 0. This makes us think that either there is a problem in the data acquisition system or a mistake was made when extrapolating the data from the database. The zero thickness slab can be regarded as an outlier and then it has been excluded

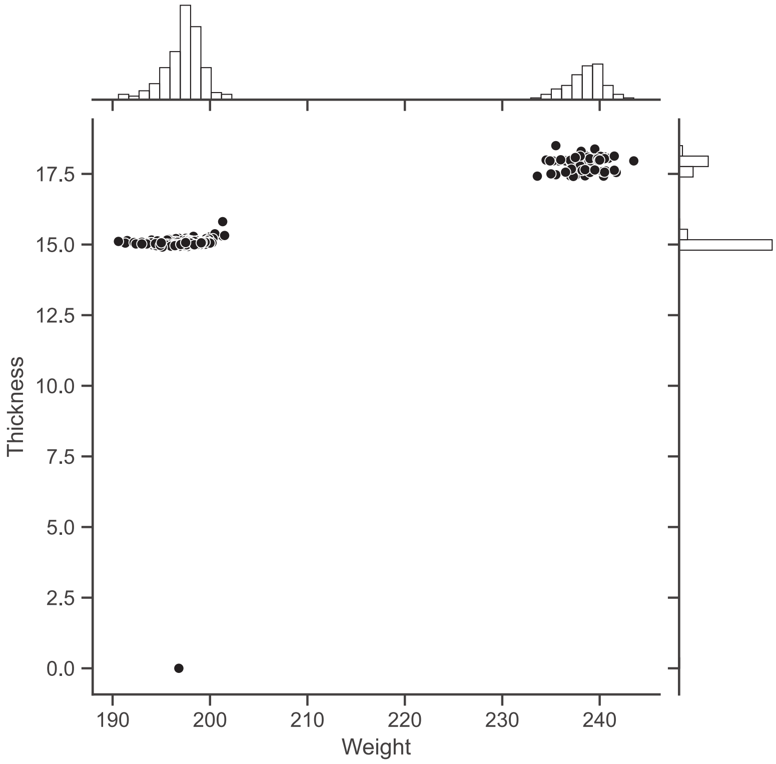


Figure 4.5: Scatter plots showing clustering of slabs.

in the following analysis. After this analysis, we have applied the IF algorithm. The algorithm associated an AS with each observation, then via a thresholding operation labeled each observation as normal or anomalous. To get a visual plot of the algorithm's predictions we have exploited PCA. In particular, we have employed PCA to get a 3-dimensional representation of the observations. Figure 4.6 and 4.7 depicts the predictions of the algorithm. From the figures

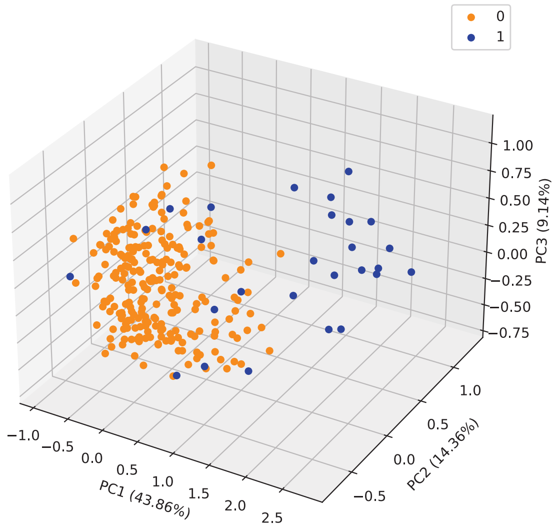


Figure 4.6: Predictions of the IF algorithm.

it is possible to notice that the algorithm has labeled as anomalous data points that are located in less dense regions. This behavior was expected from the IF model because data points located in less dense regions require a limited number of iterations to be isolated, hence they are more prone to be labeled as outliers. In figure 4.8, the 3-dimensional representation provided by PCA has been employed to show the AS, computed by IF, for each data point. Data points that have a more intense color are those considered more anomalous by the algorithm. The 5 slabs considered most anomalous with their relative AS are shown in table 4.1. Data points related to the 5 slabs considered most anomalous are highlighted in figure 4.9. The 5 slabs considered less anomalous with their relative AS are shown in table 4.2. Data points related to the 5 slabs considered less anomalous are highlighted in figure 4.10.

It is very interesting to note that the most anomalous slabs have a slab code, the unique iden-

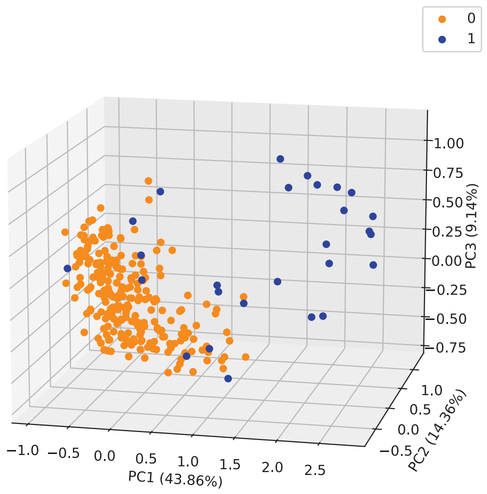


Figure 4.7: Predictions of the IF algorithm.

Anomaly Score	Slab code
$+0.212$	$SLAB01000000239464$
$+0.168$	$SLAB01000000239417$
$+0.158$	$SLAB01000000239485$
$+0.110$	$SLAB01000000239407$
$+0.102$	$SLAB01000000239463$

Table 4.1: 5 slabs considered most anomalous.

Anomaly Score	Slab code
-0.121	$SLAB01000000239622$
-0.119	$SLAB01000000239649$
-0.118	$SLAB01000000239535$
-0.118	$SLAB01000000239554$
-0.116	$SLAB01000000239537$

Table 4.2: 5 slabs considered less anomalous.

tifier of each slab, very similar to each other. Furthermore, it is possible to note that the most anomalous slabs have a slab code much lower than the less anomalous slabs. Domain experts

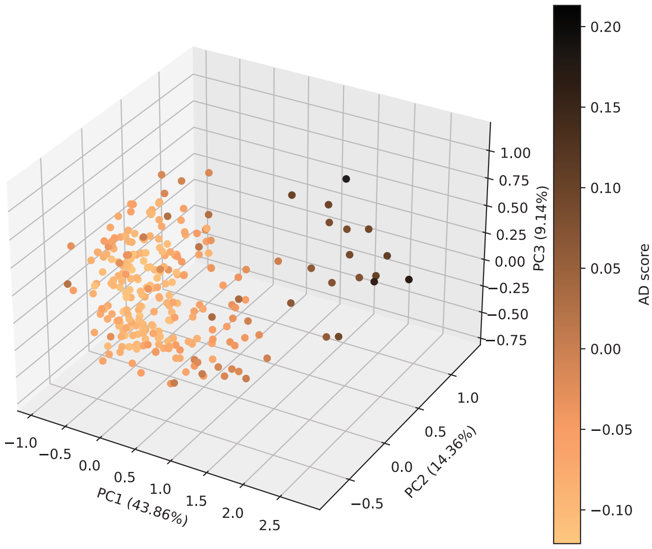


Figure 4.8: 3-dimensional representation of data points. The color intensity of the dots is proportional to the anomaly score.

had explained to us that a slab with a lower slab code than another was produced in the first. This led us to think that the first slabs produced, characterized by material composition *material code 3*, could have problems although produced by an undamaged machinery. In the light of these considerations, we have decided to visualize 4 multivariate time-series related to 4 slabs. Specifically, we have decided to display 2 multivariate time-series related to the two most anomalous slabs and 2 multivariate time-series related to the two less anomalous slabs. Figures 4.11 and 4.12 show the time-series related to the two most anomalous slabs. Figures 4.13 and 4.14 show the time-series related to the two most anomalous slabs. Visually comparing the figures, we guess that the algorithm penalizes the time-series that exhibit heterogeneous behavior. From figure 4.12 it is possible to note that the accelerometers monitor very diverse vibration levels of the machinery. Whereas from figure 4.13 the accelerometers monitor very similar vibration levels of the machinery. To get a better understanding of the underlying process, we employed DIFFI and AcME. Specifically, we used DIFFI to obtain a ranking of the features from the most important one to the least important one. Hence, DIFFI has been used to explain the model's behavior at a global level. The ranking of the features and their relative importance scores are reported in figure 4.15. From the figure it is possible to note that the 5

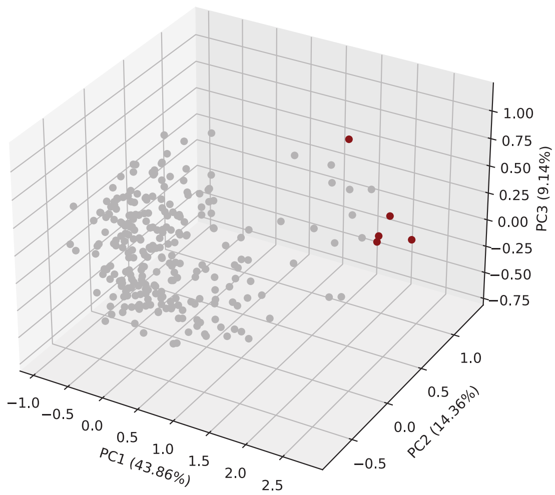


Figure 4.9: 5 data points considered most anomalous.

features considered most important by DIFFI were:

1. Fundamental frequency of the average time-series' first segment
2. Variance of the six sample means in the third segment
3. Variance of the six sample means in the first segment
4. Variance of the magnitude spectrum in a neighbourhood of the fundamental frequency of the average time-series' fourth segment
5. Variance of the magnitude spectrum in a neighbourhood of the fundamental frequency of the average time-series' second segment

This ranking provided by DIFFI supports our guess on the characteristics of the algorithm-penalized time-series because the second and third features considered most important by DIFFI are a sort of measure of the time-series heterogeneity. To further validate our guess, we employed AcME at a local level. The importance scores provided by AcME rely on perturbations of the data based on quantiles of the empirical distribution of each feature. These perturbations

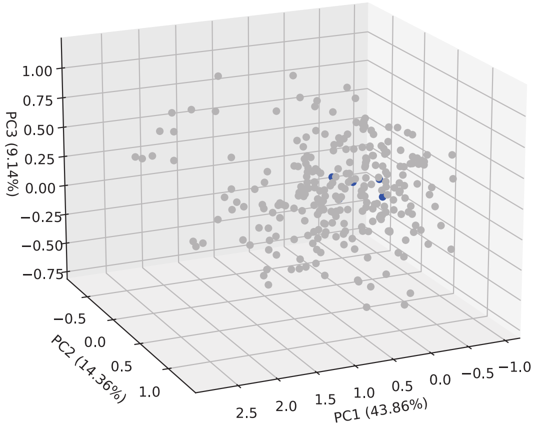


Figure 4.10: 5 data points considered less anomalous.

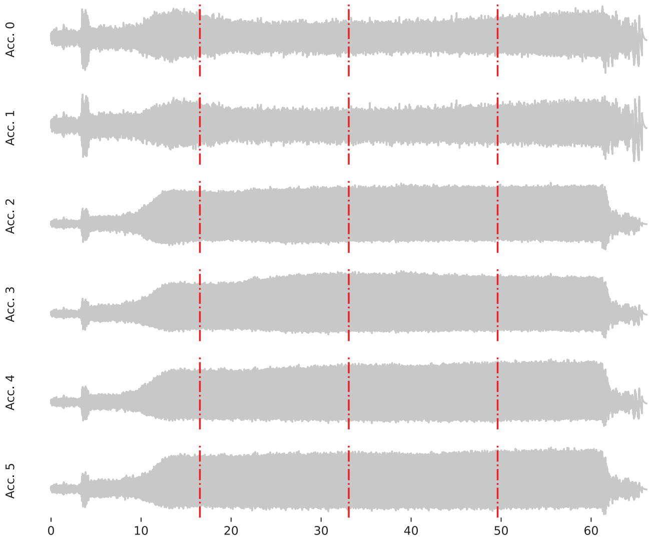


Figure 4.11: Time-series related to $SLAB01000000239464$, the most anomalous slab.

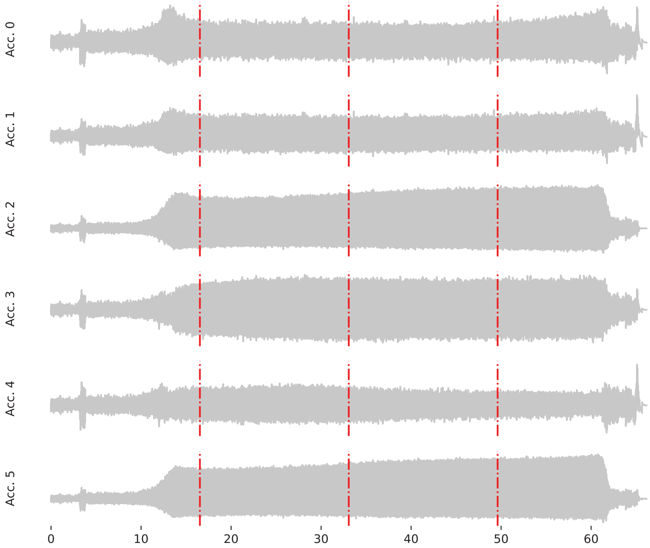


Figure 4.12: Time-series related to $SLAB01000000239417$, the second most anomalous slab.

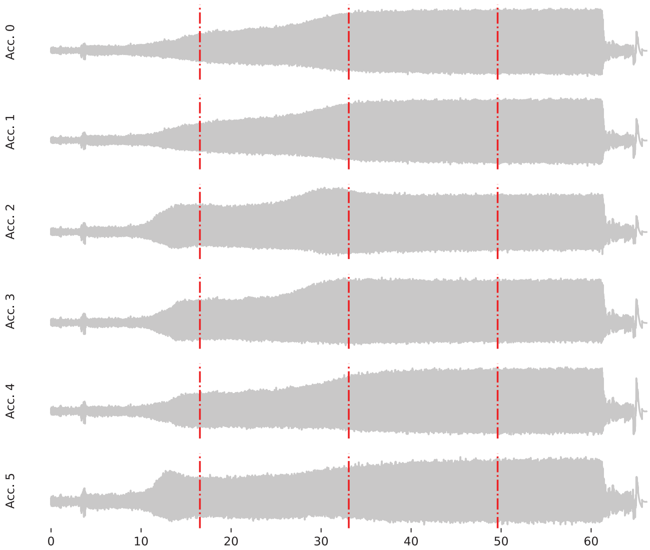


Figure 4.13: Time-series related to $SLAB01000000239622$, the less anomalous slab.

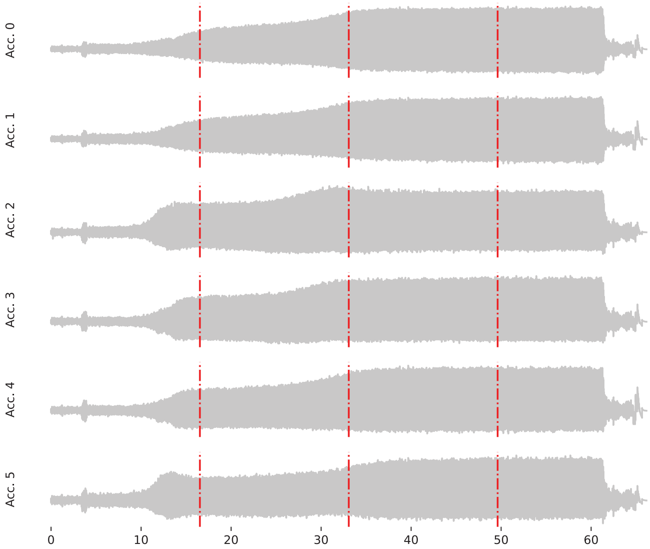


Figure 4.14: Time-series related to $SLAB01000000239649$, the second less anomalous slab.

are performed with respect to a reference point in the input space, which it is called baseline vector. When the scope of the analysis is the interpretation of individual predictions, the baseline vector is equal to the specific data point to be explained. We have decided to consider as baseline vector the feature vector related to the most anomalous slab. AcME provides similar visualizations to SHAP both for global and local interpretability. In the local case, a dashed line is placed in correspondence of the anomaly score associated with the original feature vector. Each horizontal line (associated with a specific feature) represents the actual predictions associated with the perturbed data points. AcME visualization for local interpretability makes it possible to understand how much the value of a specific feature can be reduced or increased, and how the corresponding prediction will be affected. Such a design choice allows for immediate understanding since the underlying *what-if* approach is well-aligned with human tendency to reason in counterfactual terms. In Figure 4.16 we show how local interpretability can be used as a *what-if* analysis tool. The bigger bubble corresponds to the current observation value, while the dots represent the actual predictions associated with the perturbed data points. We see that the observation with slab code $SLAB01000000239464$ has an anomaly score of 0.212, according to the IF model. Everything else remaining fixed, this score would grow over 0.23 if we increase the variance of the six sample means in the first segment or in the third segment. Namely, if the

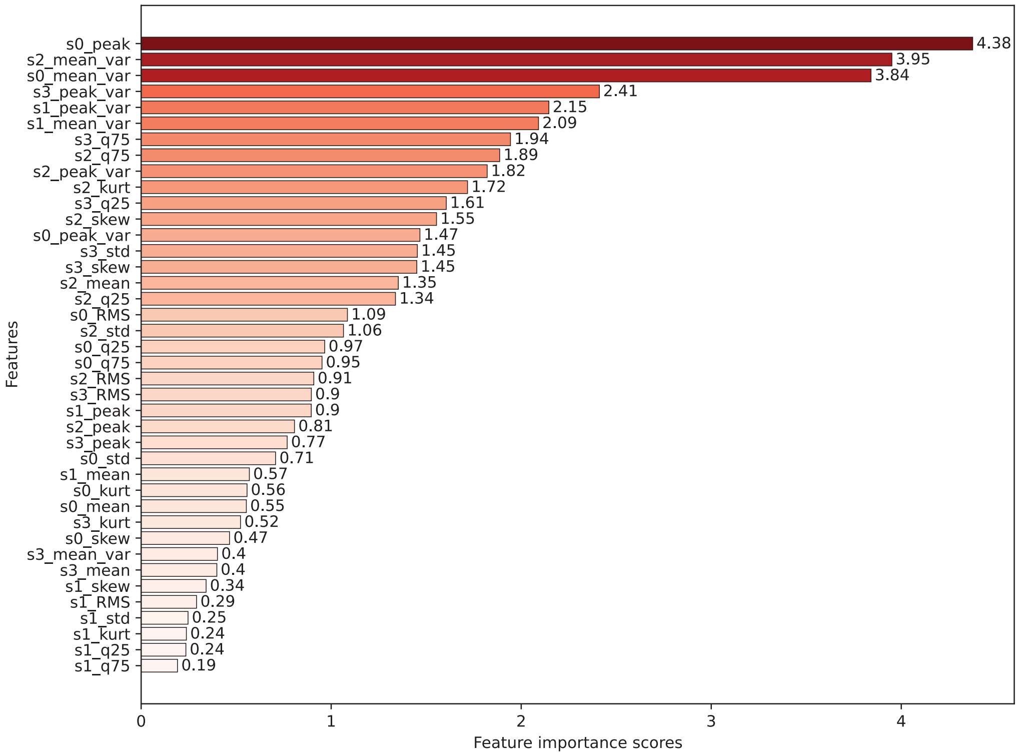


Figure 4.15: Ranking of the features provided by DIFFI.

heterogeneity of the six time-series in the first segment or in the third segment increases, the slab would be considered even more anomalous. These results support our guess on the characteristics of the algorithm-penalized time-series, that is, that the algorithm penalizes the time-series that exhibit heterogeneous behavior. In the light of these results, we have confronted ourselves with Breton domain experts that explained us that the heterogeneity of the time-series may be related to a dosing problem by the machinery that precedes, in the process of working the slabs, the vibro-compression machine. In essence, the IF model identify as anomalous slabs that have the stone particles, contained in the tank, arranged in a non-homogeneous way. Breton domain experts also explained to us that time-series' heterogeneity may be related to a mechanical failure in the machinery, hence they considered promising the proposed approach. Regarding the consideration that the most anomalous slabs have a slab code very similar to each other, Breton domain experts told us that there may be a dosing problem at the beginning of processing

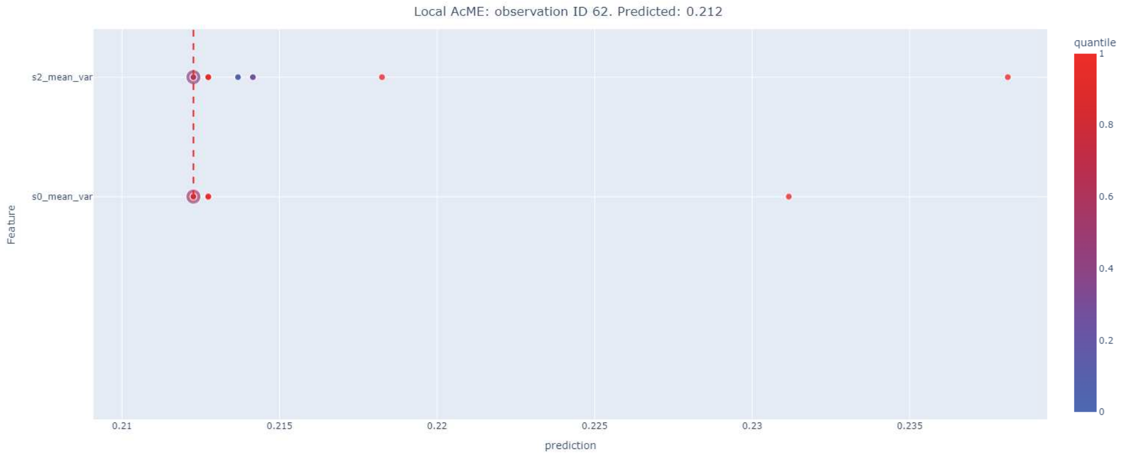


Figure 4.16: AcME local importance scores visualization for $SLAB01000000239464$.

slabs with composition *material code 3*. In figure 4.17 we have superimposed the multivariate time-series considered most anomalous by the IF model with the multivariate time-series considered less anomalous. It is possible to note that the most anomalous observation (the one in gray) has two of the six time-series with much lower energy than the less anomalous observation. This led us to think that also the average time-series of the less anomalous observation has lower energy than the average time-series of the most anomalous observation. Therefore, we made a scatter plot to display values for the ID and energy; see figures 4.18 and 4.19. We have assigned an ID to each slab, 0 for the first processed slab, 266 for the last processed slab. In figure 4.18 we have colored each observation according to the predictions of the IF algorithm, whereas in figure 4.19 we have colored each observation according to the AS provided by the IF algorithm. From these figures, it is possible to note that there is a strong correlation between ”outlierness”, energy and ID of each observation. Specifically, the slabs produced first have a more heterogeneous behavior, therefore low energy, and are consequently labeled as anomalous. These figures further support Breton's hypothesis that there is a dosing problem at the beginning of processing slabs with material composition *material code 3*.

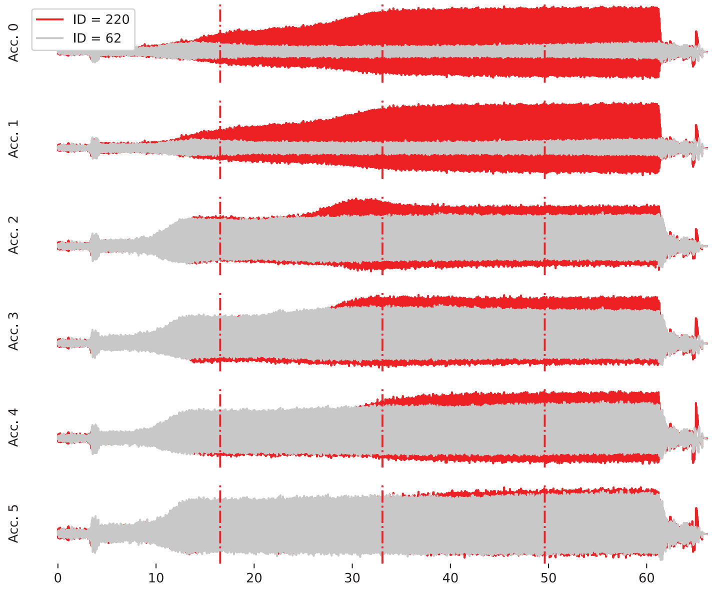


Figure 4.17: Superimposition of the multivariate time-series considered most anomalous by the IF model with the multivariate time-series considered less anomalous.

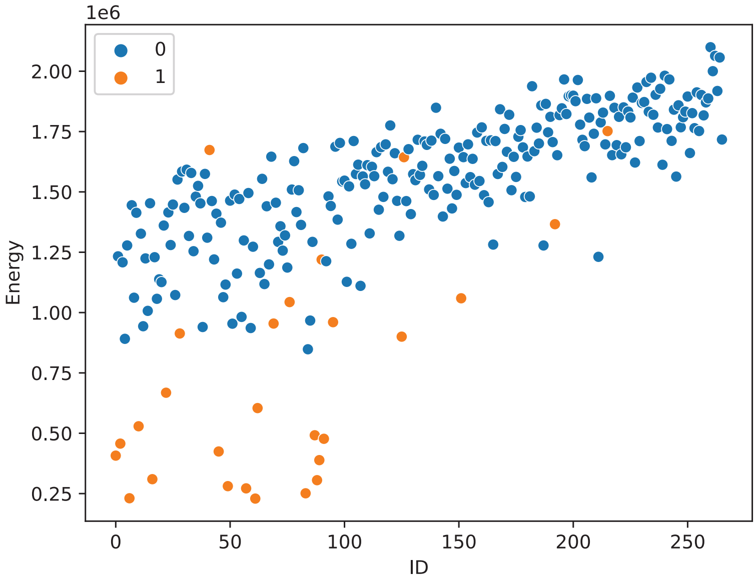


Figure 4.18: Scatter plot showing correlation between IF predictions, energy and ID.

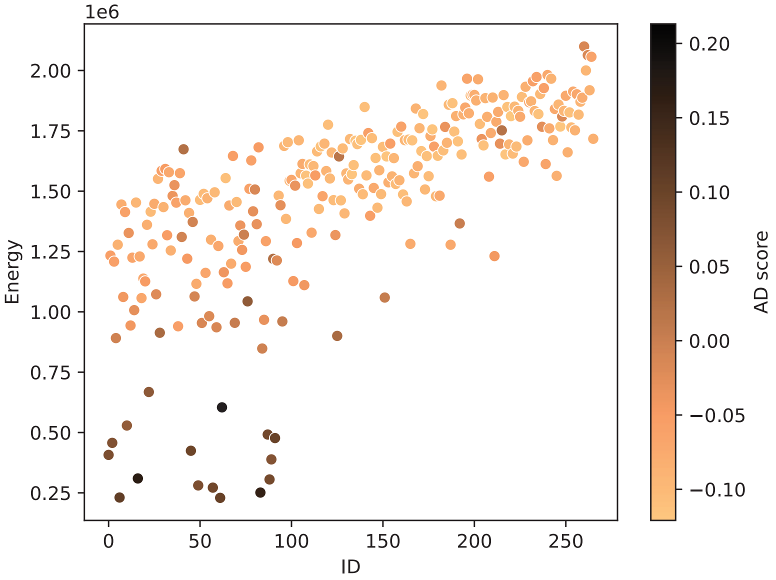


Figure 4.19: Scatter plot showing correlation between ”outlierness”, energy and ID.

4.3 Root Cause Analysis

In this section, we have adopted Root Cause Analysis to determine the root causes of two anomalous instances. To do Root Cause Analysis we exploited AcME. AcME could act as a Root Cause Analysis tool, in the sense that it may help users to figure out why the algorithm deemed a sample as normal or abnormal. In particular, exploiting the local interpretability, we could focus on an instance and use AcME to evaluate how changes in the input feature values would impact the corresponding anomaly score. Firstly, we have considered the observation with slab code $SLAB01000000239429$ that has an anomaly score of 0.001, according to the IF model; see figure 4.20. Everything else remaining fixed, this score would become negative if we decrease the RMS of the average time-series in the first segment. However, a very low value for the RMS (that corresponds to the light-blue dot on the right) would have resulted in a bigger anomaly score. When the anomaly score of an observation is negative, that observation is considered an inlier, conversely if the anomaly score is positive, the observation is considered an outlier. Hence, if the RMS of the average time-series in the first segment had been inferior, the observation would have been labeled as inlier. In figure 4.21 a superimposition of the moving RMS of the less anomalous observation (the one with ID=220) and the moving RMS of the observation considered in Root Cause Analysis is depicted. We considered a window length of $1s$ and we considered only the first segment of both signals. From the figure it is possible to note that the moving RMS of the anomalous observation is for long

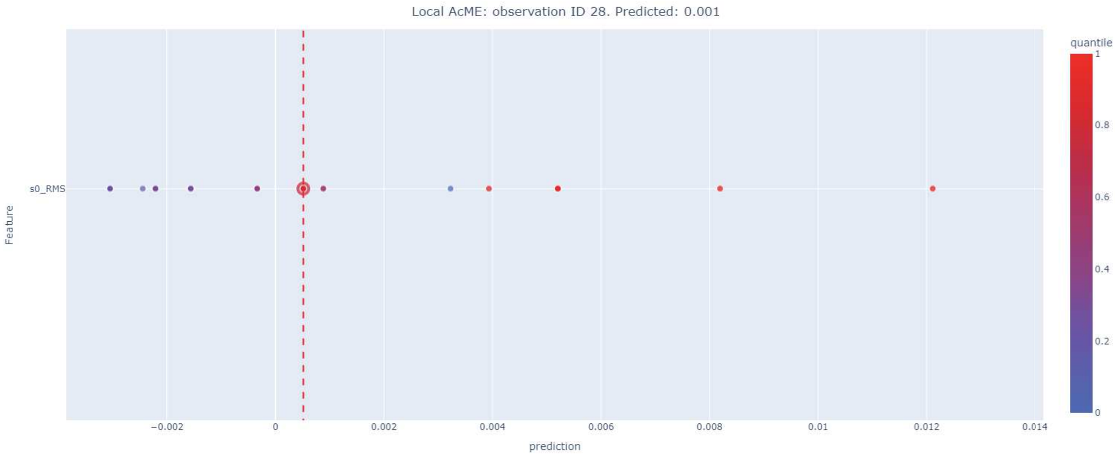


Figure 4.20: AcME local importance scores visualization for $SLAB01000000239429$.

stretches greater than the moving RMS of the less anomalous observation. This figure gave us further confirmation of the fact that by decreasing the RMS in the first segment the observation would have been labeled as inlier. Then, we examined another anomalous instance with

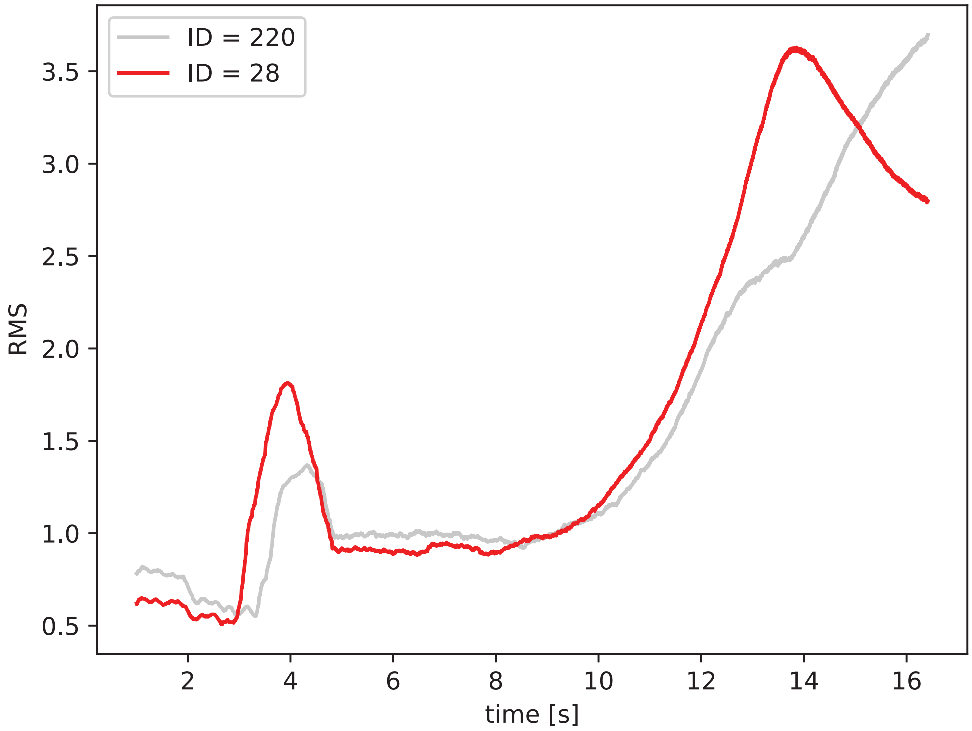


Figure 4.21: Superimposition of the moving RMS of the less anomalous observation (the one with ID=220) and the moving RMS of the observation considered in Root Cause Analysis.

slab code $SLAB01000000239664$ that has an anomaly score of 0. This is an observation with an AS equal to the IF's threshold. For this reason it is a very interesting observation. Figure 4.22 depicts AcME local scores visualization. Everything else remaining fixed, the anomaly score would become positive if we increase the variance of the six sample means in the second segment, whereas the anomaly score would become negative if we decrease the variance of the six sample means in the second segment. The bigger bubble that corresponds to the current observation value has a very intense red color, this coloring corresponds to the quantile 0.98, the rightmost dot corresponds to quantile 1, i.e. it corresponds to a bigger value, whereas the dots on the left correspond to lower quantiles, accordingly lower values. Again, it turned out that time-series' heterogeneity plays an important role in the assignment of the anomaly score by the algorithm. Precisely, if the stone particles, contained in the tank, had been arranged in a more homogeneous way the engineered stone slab $SLAB01000000239664$ would have been labeled as normal, and not as anomalous.

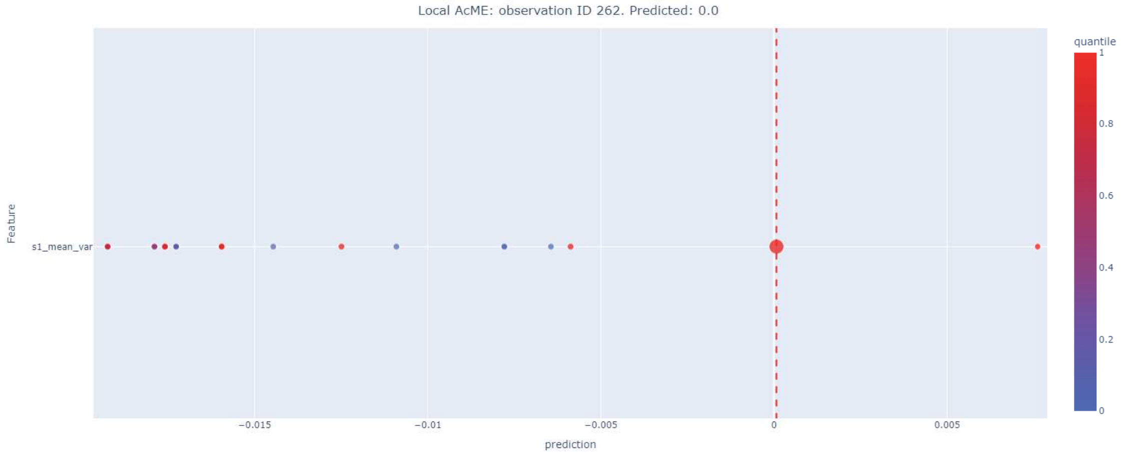


Figure 4.22: AcME local importance scores visualization for $SLAB01000000239664$.

5 Conclusions and Future Works

In this final chapter, we present the conclusions of the work, the fulfilled objectives are evaluated, and some possible future lines of work, based on the open issues identified during the project, are presented.

5.1 Conclusions

In Breton, the industrial production process of engineered stone slabs is very complex. The growing complexity of this process has generated a constant need for improvements in the safety and reliability of machinery. These two properties can be achieved through preventive actions resulting from process monitoring. Vibro-compression under vacuum represents one of the phases that make up the production process of engineered stone slabs. This thesis work presented the steps necessary to implement a Machine Learning-based approach to monitor a vibro-compression machine. The proposed approach was based on the development of a feature-based anomaly detection system. Precisely, we developed an unsupervised anomaly detection system, motivated by the lack of a reliable labelled set of anomalous data instances to describe all possible types of abnormal behaviour. Through experiments we have shown that the feature engineering process was able to identify the slabs according to their composition. This suggested us to train a separate anomaly detection model for each different material composition. We focused on the instances related to the most present material composition, specifically we trained the Isolation Forest algorithm on these instances. We focused on such an algorithm for its high detection performance, its computational efficiency and its widespread adoption. It results that the algorithm penalizes the time-series, related to the processing of a slab, that exhibit a heterogeneous behavior. This result has been supported by two interpretability methods, DIFFI and AcME, employed to provide insights on the predictions the Isolation Forest makes. Furthermore, by adopting AcME as a Root Cause Analysis tool, we have determined the root causes of an anomalous instance. The root cause resulted to be the heterogeneity of the time-series, this further supported our insights. Comparisons on the results obtained with Breton domain experts allowed us to understand that the heterogeneous behavior penalized by the algorithm was related to a dosing problem by the machinery that precedes, in the process of working the slabs, the vibro-compression machine. Accordingly, the developed anomaly detection system was able to detect slabs that have the stone particles, contained in the tank, arranged in a non-homogeneous way. For this reason the proposed approach was considered promising by domain experts. Breton experts have considered it promising also because time-series' heterogeneity could be related to a mechanical failure in the machinery. The presence of failures in vibro-compression machine may result in degradation of product quality, increased operating costs, and production stoppages which are all critical issues for manufacturing companies, including Breton.

5.2 Future Works

The work done in this thesis can definitely be extended and improved. A possible improvement of this work can be achieved through an exhaustive data collection. When we talk about exhaustive data collection we refer to the monitoring of the vibration levels of the machine for a sufficiently long time to detect anomalies and failures. Through an exhaustive data collection it would be possible to obtain a reliable labelled set of anomalous data instances to describe most of the possible types of abnormal behaviour of the vibro-compression machine. A labelled set would allow to adopt a supervised approach to the anomaly detection problem. This would be an advantage in terms of performance, as using labeled inputs and outputs, it would be possible to measure the accuracy of the Isolation Forest model and tune the right hyperparameters.

References

[1] S. Suta and S. Wattanasiriwech, "Preparation of engineered stones," in IOP Conference Series: Materials Science and Engineering, Tokyo, Japan, May 2019.

[2] S. Kang, "Sintering," Ceramics Science and Technology, pp. 141–169, 2012.

[3] U. Khalique, G. Xu, F. Liu, and C. Longting, "Machine health monitoring using artificial intelligence (AI)," in 32nd International Congress and Exhibition on Condition Monitoring and Diagnostic Engineering Management, Huddersfield, UK, September 2019.

[4] T. Von Hahn and C. K. Mechefske, "Self-supervised learning for toolwear monitoring with a disentangled-variational-autoencoder," International Journal of Hydromechatronics, vol. 4, no. 1, pp. 69–97, 2021.

[5] V. Dilda, L. Mori, O. Noterdaeme, and C. Schmitz. Manufacturing: Analytics unleashes productivity and profitability. [Online]. Available: https://www.mckinsey.com/business-functions/operations/our-insights/manufacturing-analytics-unleashes-productivity-and-profitability

[6] G. A. Susto, A. Cenedese, and M. Terzi, "Time-series classification methods: Review and applications to power systems data," Big Data Application in Power Systems, pp. 179–220, 2018.

[7] M. Roth, S. Schneider, J.-J. Lesage, and L. Litz, "Fault detection and isolation in manufacturing systems with an identified discrete event model," International Journal of Systems Science, vol. 43, no. 10, pp. 1826–1841, 2012.

[8] A. Nanopoulos, R. Alcock, and Y. Manolopoulos, "Feature-based classification of time-series data," International Journal of Computer Research, vol. 10, pp. 49–61, 2001.

[9] M. Müller, "Dynamic time warping," Information Retrieval for Music and Motion., pp. 69–84, 2007.

[10] J. Friedman, T. Hastie, and R. Tibshirani, The elements of statistical learning. Springer series in statistics, 2015.

[11] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," ACM Computing Surveys, 2009.

[12] L. Stojanovic, M. Dinic, N. Stojanovic, and A. Stojadinovic, "Big-data-driven anomaly detection in industry (4.0): An approach and a case study," in 2016 IEEE International Conference on Big Data, Washington D.C., USA, December 2016.

[13] E. Dufraisse, P. Leray, T. Benkhelif, and R. Nedellec, "Interactive anomaly detection in mixed tabular data using bayesian networks," in 10th International Conference on Probabilistic Graphical Models, Aalborg, Denmark, September 2020.

[14] H. Wang, M. Bah, and M. Hammad, "Progress in outlier detection techniques: A survey," IEEE Access, vol. 7, pp. 1–1, 2019.

[15] M. Carletti, C. Masiero, A. Beghi, and G. A. Susto, "A deep learning approach for anomaly detection with industrial time series data: a refrigerators manufacturing case study," Procedia Manufacturing, vol. 38, pp. 233–240, 2019.

[16] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.

[17] G. Calvagno and G. A. Mian, Notes on Digital Signal Processing, 2019.

[18] D. Zwillinger and S. Kokoska, CRC Standard Probability and Statistics Tables and Formulae. Chapman & Hall, 2000.

[19] E. Breck, M. Zinkevich, N. Polyzotis, S. Whang, and S. Roy, "Data validation for machine learning," in Proceedings of SysML, 2019.

[20] The what, why and how of feature engineering. [Online]. Available: https://acuvate.com/blog/the-what-why-and-how-of-feature-engineering/

[21] H. L. Shang, "A survey of functional principal component analysis," AStA Advances in Statistical Analysis, vol. 98, no. 2, pp. 121–142, 2014.

[22] K. F. Ávila Okada, A. Silva de Morais, L. C. Oliveira-Lopes, and L. Ribeiro, "A survey on fault detection and diagnosis methods," in 2021 14th IEEE International Conference on Industry Applications (INDUSCON), Virtual, August 2021, pp. 1422–1429.

[23] V. Kotu and B. Deshpande, Data Science Concepts and Practice, 2nd ed. Morgan Kaufmann, 2018.

[24] F. T. Liu, K. Ting, and Z.-H. Zhou, "Isolation forest," in Proceedings of the 2008 Eighth IEEE International Conference on Data Mining., December 2008, pp. 413–422.

[25] ——, "Isolation-based anomaly detection," ACM Transactions on Knowledge Discovery From Data (TKDD), vol. 6, pp. 1–39, 03 2012.

[26] S. Zhong, S. Fu, L. Lin, X. Fu, Z. Cui, and R. Wang, "A novel unsupervised anomaly detection for gas turbine using isolation forest," in 2019 IEEE International Conference on Prognostics and Health Management (ICPHM), San Francisco, USA, June 2019.

[27] C. Meng-ting, F. Xiao-wei, D. Zhong-hua, L. Xi, W. Xiao-long, X. Yuan-wu, and X. Tao, "Data-Driven Fault Detection for SOFC system based on Random Forest and SVM," in 2019 Chinese Automation Congress (CAC), Hangzhou, China, November 2019, pp. 2829–2834.

[28] S. Liu, Z. Ji, and Y. Wang, "Improving anomaly detection fusion method of rotating machinery based on ANN and isolation forest," in 2020 International Conference on Computer Vision, Image and Deep Learning (CVIDL), Chongqing, China, July 2020, pp. 581–584.

[29] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," Artificial Intelligence, vol. 267, no. 06, 2017.

[30] C. Molnar, Interpretable Machine Learning, 2019.

[31] A. Adadi and M. Berrada, "Peeking inside the black-box: A survey on explainable artificial intelligence (XAI)," IEEE Access, vol. 6, pp. 52 138–52 160, 2018.

[32] J. Angwin, J. Larson, S. Mattu, and L. Kirchner, "Machine bias," Propublica, 2016.

[33] A. Fabris, A. Mishler, S. Gottardi, M. Carletti, M. Daicampi, G. A. Susto, and G. Silvello, "Algorithmic audit of italian car insurance: Evidence of unfairness in access and pricing," Computing Research Repository (CoRR), 2021.

[34] K. Stubbs, P. J. Hinds, and D. Wettergreen, "Autonomy and common ground in human-robot interaction: A field study," IEEE Intelligent Systems, vol. 22, no. 2, pp. 42–50, 2007.

[35] M. T. Ribeiro, S. Singh, and C. Guestrin, “”why should I trust you?”: Explaining the predictions of any classifier,” CoRR, 2016.

[36] A. Barredo Arrieta, N. Díaz-Rodríguez, J. Del Ser, A. Bennetot, S. Tabik, A. Barbado, S. Garcia, S. Gil-Lopez, D. Molina, R. Benjamins, R. Chatila, and F. Herrera, “Explainable artificial intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI,” Information Fusion, vol. 58, pp. 82–115, 2020.

[37] A. M. Kotriwala, B. Klöpper, M. Dix, G. Gopalakrishnan, D. Ziobro, and A. Potschka, “XAI for operations in the process industry - applications, theses, and research directions,” in AAAI Spring Symposium: Combining Machine Learning with Knowledge Engineering, Stanford University, Palo Alto, USA, March 2021.

[38] M. Carletti, M. Terzi, and G. A. Susto, “Interpretable anomaly detection with DIFFI: Depth-based isolation forest feature importance,” CoRR, 2020.

[39] S. Lundberg and S.-I. Lee, “A unified approach to interpreting model predictions,” in 31st Conference on Neural Information Processing Systems (NIPS), Long Beach, USA., December 2017.

[40] D. Dandolo, C. Masiero, M. Carletti, D. Dalle Pezze, and G. A. Susto, “AcME - Accelerated Model-agnostic Explanations: Fast whitening of the machine-learning black box,” CoRR, 2021.

Acknowledgments

I would like to thank Prof. Gian Antonio Susto and Federico Milan for the opportunity they gave me and for supporting me throughout the duration of my thesis work. I really want to thank Chiara for the invaluable support she gave me during these months. She taught me a lot and she also passed on to me her passion for Machine Learning.