



Anomaly detection and event mining in cold forming manufacturing processes

Diego Nieves Avendano¹ · Daniel Caljouw² · Dirk Deschrijver¹ · Sofie Van Hoecke¹

Received: 10 August 2020 / Accepted: 24 September 2020 / Published online: 10 October 2020
© The Author(s) 2020

Abstract

Predictive maintenance is one of the main goals within the Industry 4.0 trend. Advances in data-driven techniques offer new opportunities in terms of cost reduction, improved quality control, and increased work safety. This work brings data-driven techniques for two predictive maintenance tasks: anomaly detection and event prediction, applied in the real-world use case of a cold forming manufacturing line for consumer lifestyle products by using acoustic emissions sensors in proximity of the dies of the press module. The proposed models are robust and able to cope with problems such as noise, missing values, and irregular sampling. The detected anomalies are investigated by experts and confirmed to correspond to deviations in the normal operation of the machine. Moreover, we are able to find patterns which are related to the events of interest.

Keywords Predictive maintenance · Anomaly detection · Association rule mining · Multivariate data · Matrix profile

1 Introduction

In recent years, there has been an increased interest in analysis tools for industrial applications. Within Industry 4.0, one of the goals is combining sensor technologies with data analysis tools in order to improve the manufacturing process. In general, predictive maintenance (PdM) focuses on diagnosing the machine status and providing insights about its current and future conditions. In this paper, we analyze two key aspects of PdM: online anomaly detection (AD) and event prediction on a cold forming manufacturing line.

The cold forming line is equipped with acoustic emission sensors (AE) that provide high-frequency information about the mechanical conditions of the press components. This type of sensors has previously been used to investigate failure modes of mechanical components under laboratory settings [4, 7]. In contrast, this work focuses on analyzing the signal under real operation conditions, which poses

challenges in the data quality and model capabilities. In particular, the data is challenging as samples are discontinuous and sampled at irregular intervals. This, in turn, prevents the usage of standard time series analysis approaches. Other problems which are also addressed include presence of noise, sensors getting disconnected, logging errors, and constant stops and restarts of the machines due to production bottlenecks.

Anomalies are events that do not conform to the normal or expected behavior of a process. Their correct detection is useful as it can uncover events of interest. In the manufacturing industry, AD provides benefits in terms of quality control, safety, and reduction of costs due to unexpected breakdowns. Fault prediction aims to discover early signs that an error will occur and to report them in terms of probability of the error occurring or a time interval in which it will occur. Fault prediction provides benefits, as it enables the optimization of maintenance schedules.

Traditional AD algorithms use statistical approaches to find extreme values or correlation changes between features [9]. This leaves a considerable gap in scenarios where anomalies are not necessarily linked to extreme values, or where their statistical properties do not diverge considerably from the norm. There exists extensive literature in the detection of anomalies in online settings [1] and multivariate cases [9], but their application in industrial settings is still limited. More specifically, an unsupervised AD approach based on the micro-cluster continuous outlier detection

✉ Diego Nieves Avendano
diego.nievesavendano@ugent.be

Daniel Caljouw
daniel.caljouw@philips.com

¹ IDLab, Ghent University - imec, Ghent, Belgium

² Philips Consumer Lifestyle B.V., Eindhoven, Netherlands

(MCO) algorithm [13] is evaluated on the same industrial use case as of the current work [18].

The AD approach is based on matrix profile (MP) [22], which is a method for time series analysis that is robust, domain-agnostic, and computationally efficient. Compared with previous approaches, it offers the following advantages:

- Requires a short warm-up time and no historical data to give reliable results.
- Provides a bounded and intuitive anomalous score.
- Can be calculated in real-time as it is not computationally intensive.
- Can handle non-uniform sampling.

Fault prediction in industrial applications using wave signals, such as vibration and acoustic emission, has been applied for valves [2], reciprocating machines [6], diesel engines [5], indenters [19], conveying systems [20], and other rotating components [20, 21]. Traditional approaches focus on the analysis of temporal and frequency features in combination with statistical testing. More recently, data-driven methods such as self-organizing maps [21], support vector machines [2], and deep learning [14] have been investigated. In our approach, we present a model for event prediction, where event refers to not only fault events, but also events which may not be associated to an error but are of interest for PdM.

The event prediction is based on a combination of salient subsequence mining, clustering, and association rules. It is generic enough to deal with different types of sensor data, with non-uniform sampling, and requires few fine-tuning of parameters. Compared with previous work, we are handling data in industrial settings in comparison with approaches done under controlled experimental settings. In our literature review, only the work from Varga et al. [20] handles real industrial conditions, but for a different type

of machinery and without classifying the different types of faults. To the best of our knowledge, we are the first to present methods using AE for PdM in cold forming lines. State of the art machine learning techniques rely on data abstractions which reduce interpretability. In recent years, there has been an increased demand of interpretable models within different industrial sectors [12]. By using association rules, the acoustic patterns which signal future faults can be easily retrieved.

This paper is organized as follows: Section 2 presents the use case and the data representation. Section 3 presents the methodology for the anomaly detection based on matrix profile, and the event prediction based on association rules. Section 4 presents the results. Section 5 discusses the results and the lessons learned. Finally, Section 6 contains the conclusions.

2 Cold forming process

The real-world use case investigated in this work corresponds to a Philips' manufacturing line in the Netherlands. The data belongs to a cold forming line of a consumer lifestyle product. Figure 1 shows a scheme of the data collection.

The data were collected under normal operation conditions for a period of over a year. The data consist of information of the material batch, sensors from the press module, and information from the preventive and corrective maintenance. Table 1 summarizes the components and the information retrieved.

The press module is composed of a main rotor and 6 dies which either cut or flatten the metal strip. The dies of interest contain acoustic sensors or position sensors, or both. The position sensors are located within the lower die and measure the correct position of the stripper plate.

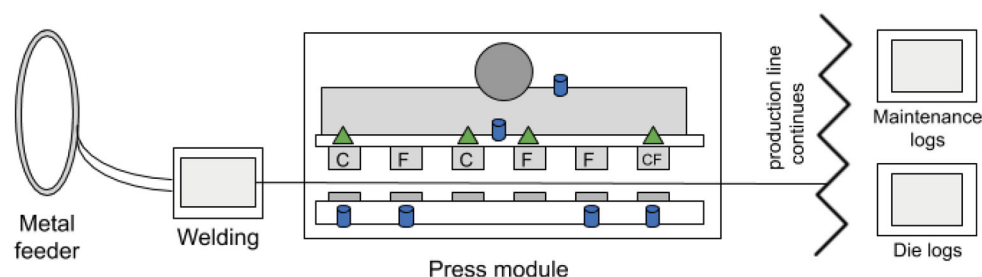


Fig. 1 Cold process diagram. The blue cylinders correspond to the location of the acoustic emission sensors; the green triangles correspond to the location of the position sensors. Each die inside the press module can either cut (C), flatten (F), or both (CF). After the press

module, the production line continues with further steps for the final product. Separately, the maintenance and die logs provide information concerning the status and maintenance of the press components

Table 1 Manufacturing stages and information retrieved

Stage	Information
Metal feeder	Reel ID
Press	Die position at 7 locations
	Acoustic emission at 6 locations
Maintenance	Die logbook
	Maintenance logbook

The stripper position deviates when there is scrap material inside the die. In order to prevent damage, the position sensor triggers a stop of the machine. The AE sensors are located in the proximity of the dies; they measure the acoustic waves propagated through the die. The goal of this work is to analyze the AE signal to provide insights of the machine status. Table 2 summarizes the die functions and the available sensors.

2.1 Acoustic emission signal

The cold forming press contains six AE sensors. These sensors record sound information along a wide frequency spectrum. Changes in the AE signal can be associated with gradual mechanical changes, such as those caused by friction and mechanical degradation, as well as sudden changes caused by unexpected events such as jamming of materials.

The acoustic emission is measured in a low-frequency band (< 2 kHz), filtered and then a temporal feature is extracted for specific positions of the main rotating component. The temporal feature extracted is not disclosed for intellectual property reasons. Figure 2 shows common patterns for each of the six sensors. The x -axis is not time but the angle of the main rotor. In normal operating conditions,

Table 2 Press components and sensors

Component	Acoustic sensor name	Die type	Position sensors
Ram	RAM AE	–	–
Housing	G0 AE	–	–
First die	G2 AE	C	Yes
Second die	G3 AE	F	No
Third die	–	C	Yes
Fourth die	–	F	Yes
Fifth die	G4 AE	F	No
Sixth die	G7 AE	C F	Yes

C cutting, F flattening

the speed is constant and the angle position is approximately equivalent to time. The magnitude (y -axis) is directly related to shear and tension applied on the sensor's main axis and has an arbitrary unit.

In order to reduce the data size, only one sample per minute is kept. Due to limitations in the sensing equipment, the samples are not uniformly distributed. The median difference between subsequent samples is of 63 ± 4.44 s (median and median absolute deviation). Regardless of the non-uniform sampling, our approach is flexible and does not require the signal to be continuously measured.

2.2 Stop and maintenance events

The press module is equipped with sensors that automatically stop the press in case of faulty condition(s). The stops can be triggered by either lubrication problems or the presence of double material. The double material can be detected in four modules as described in Table 2.

Additionally, there are maintenance events which are performed after faults in the press or deviations in the product quality have been detected. The maintenance events are associated to specific die modules and reported manually. We focus on two types of maintenance events: scrap returns into the die and cutting faults that occur due to a broken punch or a blunt punch.

In total there are four different types of events of interest: two stop events and two maintenance events. For the rest of this paper, we will refer to them as events of interest regardless of their type.

3 Methodology

This section presents the methodology for the AD model and the rule mining model. Figure 3 shows the data flow and processing. First, the acoustic emission is converted into the meta-time series using the matrix profile as described in Section 3.1; this information is then processed in two branches: the AD model which operates online, described in Section 3.2, and the event prediction which is conformed by three blocks which are described in Section 3.3. Once the anomalies have been detected or labelled, they can be passed to the rule mining module to learn patterns between anomalies and events of interest.

3.1 Online matrix profile

The MP is a data mining technique that allows analysis of time series in an intuitive way [22]. It focuses on the

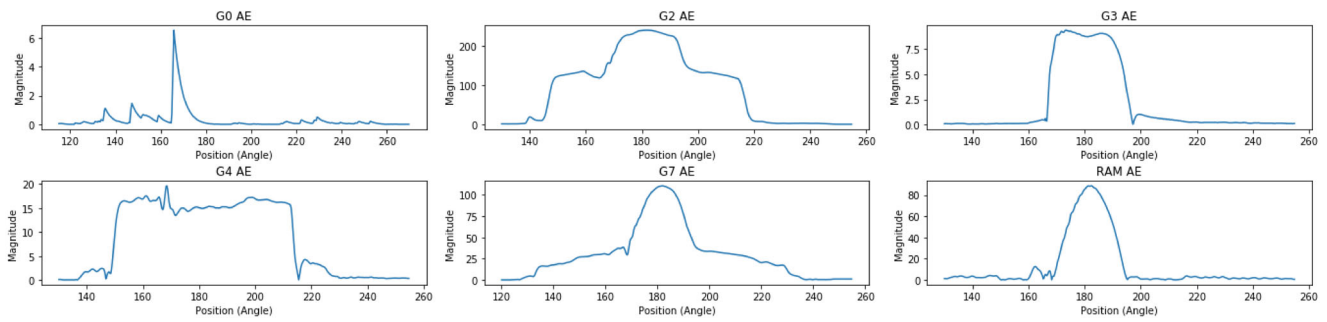


Fig. 2 Patterns corresponding to each of the acoustic sensors

problem of all-pairs-similarity-search, where the nearest neighbor for each object in a set is found. The algorithm requires only two parameters: a window length (m) and a distance function. The parameter m is defined as the same number of angles positions ($m = 500$). The distance function used is the z -normalized Euclidean distance, which is a preferable measure when interested in the pattern shape regardless of the magnitude. The output of the algorithm are two meta-time series, namely the MP, which contains the distance to the nearest neighbor, and the profile index (PI), which contains the position of the nearest neighbor.

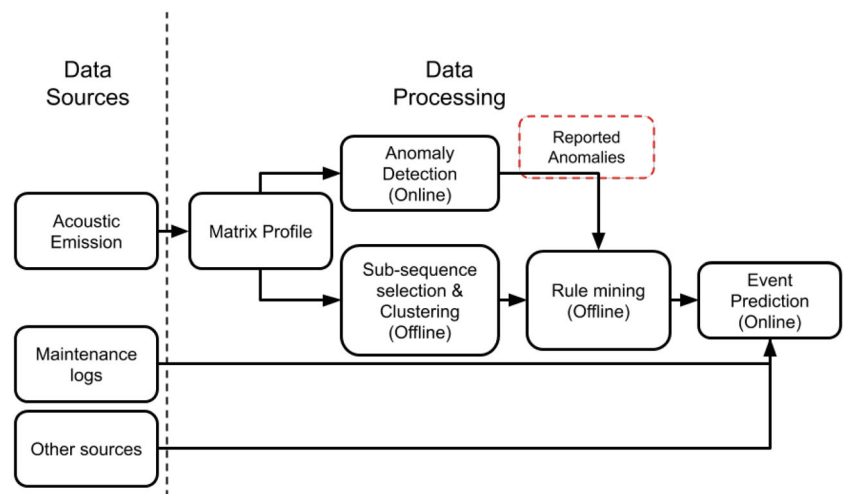
Traditional MP requires time series to be continuous and uniformly sampled. In this use case, the signal is subsampled at non-uniform intervals; and therefore, the problem is posed in a different representation: the time series X consists of D dimensions, each dimension corresponding to an AE sensor, N samples, each sample corresponding to a full punch recording, and M attributes corresponding to the magnitude at each angle. Hence, $X \in$

$\mathbb{R}^{D \times N \times M}$. In this case, $D = 6$, $M = 500$. A new sample $x \in \mathbb{R}^{6 \times 1 \times 500}$ arrives every minute approximately.

We modify the original matrix profile in the following aspects:

- The exclusion zone corresponds to the number of full punches to ignore. This differs from the original approach where the exclusion zone corresponds to the number of time steps to exclude.
- Only the left MP is calculated, which corresponds to the comparison of the current sample with the previously received samples. This allows the method to be used in online settings.
- We define a bound value to limit the number of previous samples to which an incoming sample is compared. The bound is set to 1800 samples for all the experiments, which corresponds to approximately the previous 30 h. This reduces the computation time while still giving a good approximation of the MP without reducing the AD capabilities.

Fig. 3 Data sources and processing for anomaly detection and association rules models



Algorithm 1 presents the code for computing the online MP and PI.

Algorithm 1 Algorithm to obtain the matrix profile and profile index.

1 **matrix profile** (*ts*, *bound*, *e*, *pi*, *mp*, *buffer*)

Input : A new time series subsequence **ts** with **D** dimensions and length **m**, a bound limit value **bound**, an exclusion value **e**, the matrix profile **mp**, the profile index **pi** and the buffer array **buffer** containing the previous subsequences

Output: Incrementally update the matrix profile, profile index and buffer

```

2 foreach d in D do
3   sequences ← buffer[d]
4   mindist ← inf
5   minix ← -1
6   for ix from 0 up to bound - e do
7     dist ← distfun(ts[d], sequences[ix])
8     if dist < mindist then
9       mindist ← dist
10      minix ← ix
11   if length(buffer) > bound then
12     buffer[d] ← [buffer[d][1:], ts]
13   else
14     buffer[d] ← [buffer[d], ts]
15   mp[d] ← [mp[d], mindist]
16   pi[d] ← [pi[d], minix]
17 return mp, pi, buffer

```

Each channel is processed independently and we obtain a D -dimensional meta-time series where each dimension corresponds to the MP of a sensor. Using the timestamps of the original time series, the PI can be converted into time differences. This helps reduce the discrepancy caused by the non-uniform sampling. For example, an anomaly can be explained if a large time gap is found, which occurs when the machine is stopped for inspection. These new meta-time series are used for AD and rule mining as explained in the next sections.

3.2 Anomaly detection

In this work, we focus on point anomalies, where a full punch is labeled as an anomaly if any of its dimensions is anomalous. This can be done by analyzing the MP of

each channel and defining a threshold value. Additionally, the MP can be summed up over the dimensions and detect AD over the reduced MP [24]. The z -normalized Euclidean distance is bounded between $[0, m]$; therefore, the multidimensional MP can be summed up regardless of the magnitude of the original signals. In case one wants to give more focus to a specific channel, the sum can be weighted.

In this use case, the anomalies are unlabelled. Therefore, a threshold value in the MP cannot be defined in a straightforward way. Instead, a statistical approach is used. The running mean and standard deviation over each channel of the MP and the summed MP are calculated and the threshold bands are defined as the mean $\pm n\sigma$. With n , the number of deviations to consider as the tolerance limit. A punch is considered as anomalous if its MP value is beyond the bands. Figure 4 shows the moving average and the $\pm 6\sigma$ bands for a period of 8 days. The gaps in the plot are caused by stops in the machine. When the machine is restarted, the MP will be high due to lack of samples to compare against. To avoid labelling these events as anomalous, the rolling mean is only considered after at least 10 samples have arrived.

3.3 Rule mining

This section explains the rule mining approach, which aims at finding acoustic patterns which precede events of interest. It is organized as follows: Section 3.3.1 presents the salient subsequence mining, which reduces the dataset to the most representative patterns. Section 3.3.2 presents the clustering model which is applied to the reduced dataset to discretize the acoustic signals. Finally, Section 3.3.3 explains the procedure to learn association rules and construct a classifier by association.

3.3.1 Salient subsequence mining

Salient subsequence mining focuses on finding a subset of the data that represents the majority of the common patterns. We apply a salient subsequence mining technique that uses the MP to explore candidates and add them to a representation set using minimum description length (MDL) as a stop criterion [23].

The MDL technique requires the time series to be discrete; therefore, each dimension is normalized to have $\mu = 0$ and $\sigma = 1$, and afterwards discretized as shown in Eq. 1, where b is the number of bits for discretization. It has been proven empirically that the discretization of time series does not have an impact on classification and other related

Fig. 4 Matrix profile results for 2 weeks of data. Highlighted in red, the reported anomalies. The y-axis is the z-normalized Euclidean distance (unitless)

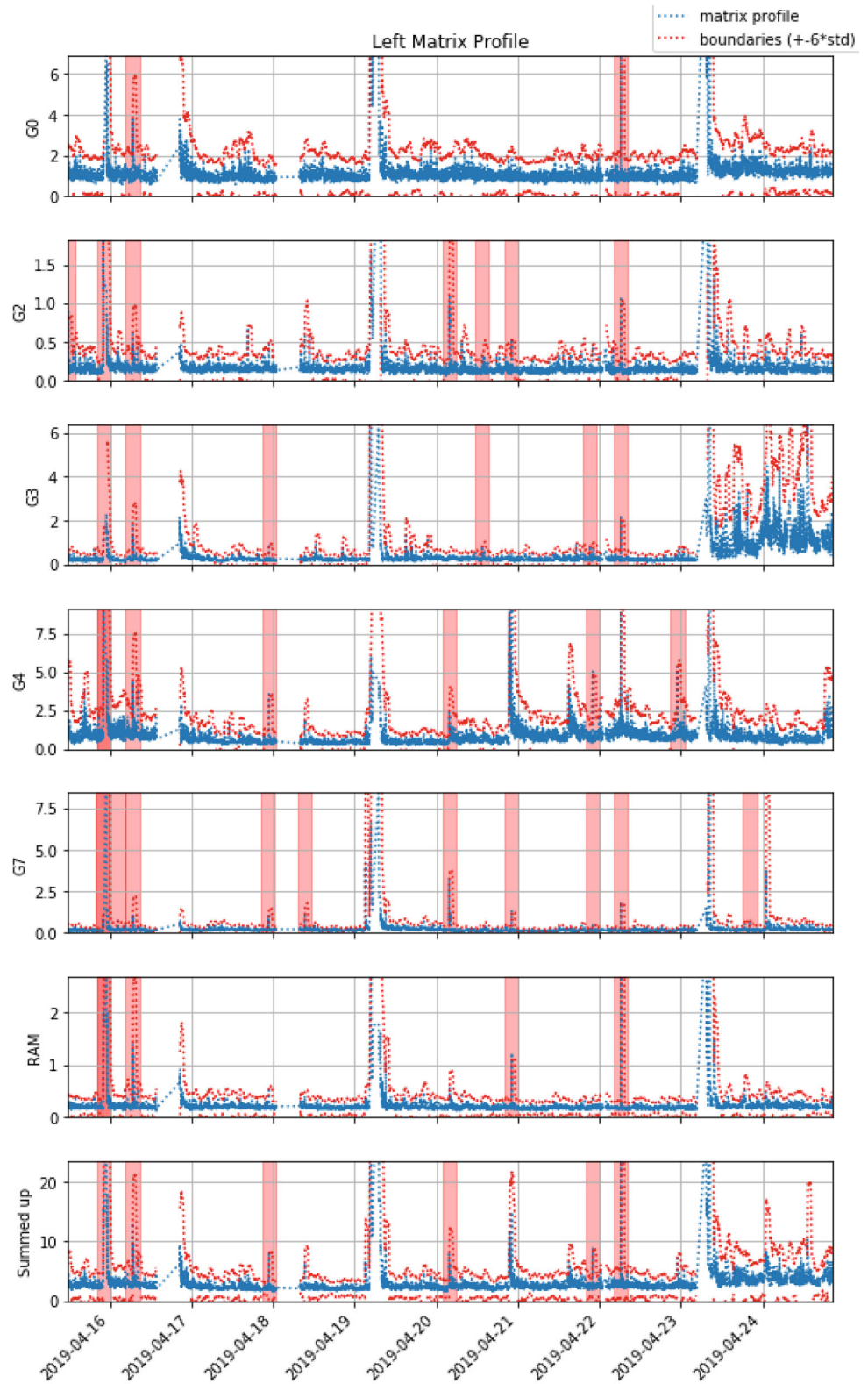
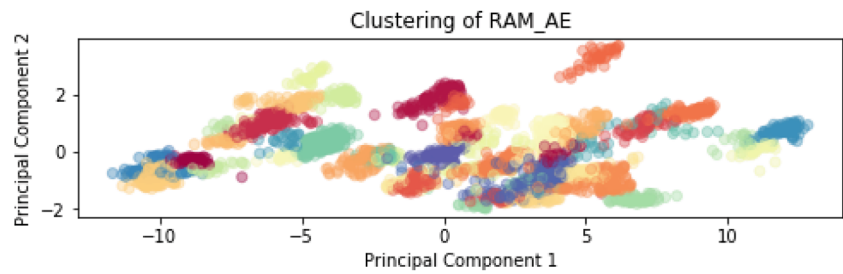


Fig. 5 Clusters for the ram. Each color represents a different cluster. In this case, the representation has well-defined clusters



tasks, and that in general the number of compression bits (b) does not have a drastic impact [11].

$$x_{discrete} = \text{round} \left(\left(\frac{x_{normalized} - x_{min}}{x_{max} - x_{min}} \right) (2^b - 1) \right) + 1 \tag{1}$$

The MP is traversed to explore the sequences that have a low value, meaning that there exists at least another closely related sequence. Notice that our MP approach includes a limited search (defined by the bound limit, see Section 3.1), which means the candidates are local minima and the solution is not optimal. However, the problem of optimal subsequence selection is on its own intractable, but good results have been obtained with approximate solutions in similar tasks [23].

Each subsequence is added to a hypothesis set, which is used to encode other segments; or to a compression set, meaning that it can be expressed by a member of the hypothesis set in fewer bits. Equation 2 shows the description length measure (dl) for a segment x , i.e., the number of bits required to store a sequence of length m with a cardinality of b bits.

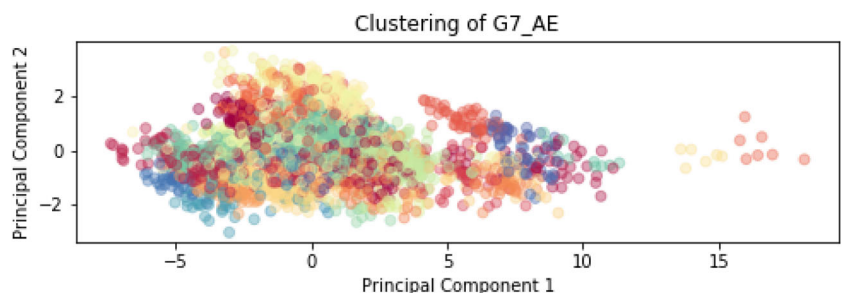
$$dl(x) = m \times b \tag{2}$$

Equation 3 shows the bit save obtained by compressing a sequence t_c with t_h as its hypothesis.

$$bit_{save} = \gamma(t_c, t_h) \times (\log_2 m + b) \tag{3}$$

where $\gamma(\cdot)$ is function that counts the number of differing digits between two discretized sequences.

Fig. 6 Clusters for the G7 channel. Each color represents a different cluster. The clusters are not clearly defined



Algorithm 2 Sequence selection.

```

1 sequence selection ( $ts, mp, pi, c$ )
   Input : A tensor of time series subsequences  $ts$  with
            $D$  dimensions and length  $m$ , the matrix profile
            $mp$  and profile index  $pi$  of  $ts$ , a number of
           bits to compress  $b$ , the number of candidates
            $n_c$  to consider each iteration and the number
           of samples to ignores as trivial matches  $n_t$ 
   Output: A reduced set of subsequences
2  $C \leftarrow \emptyset, H \leftarrow \emptyset$ 
3  $ts \leftarrow \text{normalize}(ts)$ 
4  $ts \leftarrow \text{reshape}(ts)$ 
5  $ts \leftarrow \text{discretize}(ts, c)$ 
6  $U \leftarrow$  set of all indices of  $ts$ 
7  $bcsf \leftarrow \text{calculate\_cost}(C, H, U)$ 
8 while True do
9    $cands \leftarrow \text{pick\_cands}(mp, n_c, n_t)$ 
10  if  $cands$  is empty then
11    break
12   $cand, type \leftarrow \text{pick\_best\_cand}(cands, H, pi)$ 
13  remove  $cand$  and the trivial matches from  $mp$ 
14  if  $type$  is hypothesis then
15     $H \leftarrow H \cup \{cand\}$ 
16  else
17     $C \leftarrow C \cup \{cand\}$ 
18     $bit_{cost} \leftarrow \text{calculate\_cost}(C, H, U)$ 
19    if  $bit_{cost} > bcsf$  then
20      break
21    else
22       $bcsf \leftarrow bit_{cost}$ 
23  remove  $cand$  from  $U$ 
24 return  $C \cup H$ 
    
```

Algorithm 2 presents the sequence selection which follows closely the one proposed by Yeh et al. [23], with the difference that in this case the patterns are multidimensional and the meta-time series, MP and PI, have a limited search. To employ the same algorithm, the sum of the multidimensional MP is used to select the best candidates. Alternatively, the patterns for each individual channel can be found and then later combined. However, for multidimensional time series, it is often the case that common patterns are found in a lower dimensionality [24].

Algorithm 3 Candidate sequences selection.

```

1 pick cands ( $mp, n_c, n_t$ )
   Input : A matrix profile  $mp$ , the number of candidates
            $n_c$  to select and the number of neighboring
           samples considered as trivial matches  $n_t$ 
   Output: The indexes of the candidate sequences
2  $candidates \leftarrow \emptyset$ 
3 for  $i \leftarrow 1$  to  $n_c$  do
4    $ix \leftarrow get\_min\_ix(mp)$ 
5    $candidates \leftarrow candidates \cup \{ix\}$ 
6   remove  $ix$  and the trivial matches  $n_t$  from  $mp$ 
7 return  $candidates$ 

```

Algorithm 4 Best sequence selection.

```

1 pick best cand ( $cands, H, pi$ )
   Input : A time series  $ts$ , a set of candidate indexes
            $cands$ , the hypothesis space  $H$  and the profile
           index  $pi$ 
   Output: The best candidate and its type
2  $best\_cand \leftarrow \emptyset, type \leftarrow \emptyset$ 
3  $best\_bit\_save \leftarrow -inf$ 
4 foreach  $c$  in  $cands$  do
5    $cand\_ts \leftarrow ts[c]$ 
6    $nn \leftarrow get\_nn(c, pi)$ 
7    $bit\_save \leftarrow bitsave(nn, cand\_ts)$ 
8   if  $bit\_save > best\_bit\_save$  then
9      $best\_bit\_save \leftarrow bit\_save$ 
10     $best\_cand \leftarrow c$ 
11     $type \leftarrow "hypothesis"$ 
12  foreach  $hypo$  in  $H$  do
13     $bit\_save \leftarrow dl(cand\_ts) - rdl(cand\_ts|hypo)$ 
14    if  $bit\_save > best\_bit\_save$  then
15       $best\_bit\_save \leftarrow bit\_save$ 
16       $best\_cand \leftarrow c$ 
17       $type \leftarrow "compressible"$ 
18 return  $best\_cand, cand\_type$ 

```

The sequence selection goes through the following stages: (1) each dimension is normalized in order for them

to have comparable magnitudes, (2) the data is reshaped from a tensor of D dimensions with m instances per sample to 1 dimension and $D \times m$ instances per sample (lines 3–5). The initial cost (line 7) is equivalent to storing all the sequences without compression. Then, a greedy search loop starts, where first a set of candidates is selected using the MP (line 9 in Algorithm 3), and then evaluated to determine the best pattern in terms of *bits saved*, and whether it should be assigned to the hypothesis or compression set (line 12 in Algorithm 4).

Although one could argue that the subsequence selection is not required as clustering can be done on the whole dataset, it is important to consider that reducing the dataset decreases considerably the computational power required. In addition, it has been proven that clustering time series becomes irrelevant if all possible subsequences are considered [11].

3.3.2 Clustering

The salient subsequences discovered in Section 3.3.1 are clustered by first using principal component analysis (PCA) and then applying Gaussian mixture clustering on the principal components (PC). The aim is to use the discovered clusters as labels for the AE signals.

The PCA projection is done as follows: first, each dimension is considered to be a stationary time series, and is normalized to have $\mu = 0$ and $\sigma = 1$. Then, for each punch, the sample is seen as the concatenation of six channels over a full rotation ($d \times 500$ samples) giving a total of 3000 instances per punch in this case.

The clustering is done using Gaussian mixture models which is a family of probabilistic model that represents the data as a weighted sum of normal distributions, each of them with its own covariance matrix. Although any clustering algorithm can be used, the Bayesian Gaussian mixture algorithm offers the advantage of reducing the number of clusters in case it is required [3].

We analyze the effects of clustering on individual dimensions. Figure 5 shows the projection in the first two components for the ram. Here, it can be appreciated that the data has some defined clusters. Figure 6 shows the clustering for component G7, where the clusters are not clearly defined. Figure 7 shows the projection of the 6 dimensions combined.

3.3.3 Association rules

The association rule problem focuses on discovering interesting relations in datasets with transactions. This problem has been of particular interest in the retail industry, where associations can be found between products that are bought together or subsequently. We are interested in the

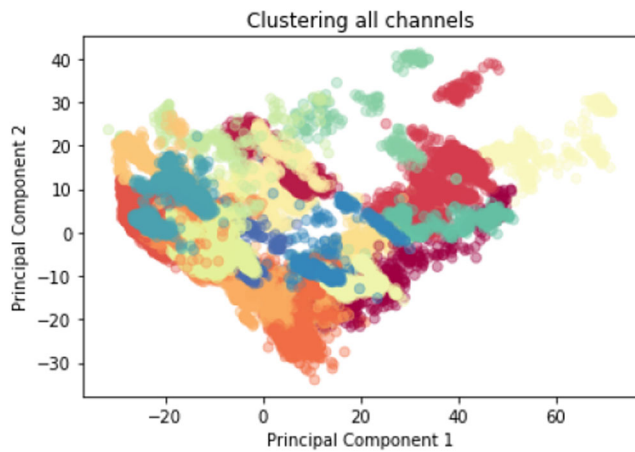


Fig. 7 Clusters discovered for the combined dimensions. Each color represents a different cluster

association rules approach as it provides associations with conditional probabilities, which facilitate the interpretability of the results. The problem is as follows: given a set of transactions \mathcal{D} where each transaction \mathcal{T} is a set that contains a number of items i from the set $\mathcal{I} = \{i_1, i_2, \dots, i_m\}$. A transaction \mathcal{T} contains the set of items \mathcal{X} if $X \subseteq T$. Then, an association rule is stated as follows: $X \Rightarrow Y$, where $X \subseteq I, Y \subseteq I$ and $x \cap y = \phi$. Which can be read as follows: given X, Y is more likely to occur. In our model, X corresponds to a set of one or multiple acoustic signals discretized by the clusters, and Y is an event of interest.

The frequent pattern growth (FP-growth) algorithm is a fast algorithm for frequent pattern mining which uses an efficient data representation to handle large databases. It is particularly useful in cases where the desired rules contain sets of items with low support threshold. This coincides with finding rules that predict the events of interest. The full specifications of the FP-growth algorithm can be found in the work by Jiawei et al [10].

The problem then can be seen as mining item sets from the combined list of acoustic clusters and events. In this case, the amount of unique items is relatively low, with few possible clusters and only four event types (as explained in Section 2.2).

The AE data is segmented in time windows with no overlap. Each pattern within the window is encoded according to the clusters discovered in Section 3.3.1 and combined with the maintenance logs to form a transaction \mathcal{T} . Notice that the acoustic clusters must precede the maintenance events, but we are not interested in the order in which the patterns occur. In case no events occur within the window, a dummy healthy status is assigned to the transaction.

The generated rules are filtered to only keep the ones with AE clusters in their antecedents and events of interest as their consequent. In addition, only the rules with a

minimum confidence (as defined in Formula 4) and lift (as defined in Formula 4) are kept. *Support* is the percentage of transactions in \mathcal{D} which contains the given item or tuple of items. The lift of a rule is the ratio between the probability of the antecedent and consequent occurring together against the expected probability if they were independent. This is based on $P(A \cap B) = P(A)P(B)$, where the probability of A is not affected by B nor the other way around.

$$Confidence(X \rightarrow Y) = \frac{Support(X \cup Y)}{Support(X)} \tag{4}$$

$$Lift(X \rightarrow Y) = \frac{Confidence(X \cup Y)}{Support(Y)} \tag{5}$$

Finally, the association rules are grouped in order to form a classifier by association using a modified version of the CBA-CB algorithm [15]. In the original implementation, rules are sorted according to their confidence and support. We extend this by taking into consideration the class imbalance by adding a weighted confidence which is the confidence of a rule multiplied by the inverse frequency of its consequent. Algorithm 5 presents the methodology to create a classifier based on association rules.

The algorithm first sorts the generated rules as follows: Given the rules r_i precedes r_j if:

- Confidence of r_i is larger than that of r_j
- If their confidences are equal but support of r_i is larger than r_j
- If both confidences and supports are equal but the weighted confidence of r_i is generated before r_j
- Else the rule that was generated first.

Table 3 Anomalies discovered as % of the punches for two different band levels

	Anomalies (%)	
	$\pm 3\sigma$	$\pm 6\sigma$
G0	1.363	0.035
G2	1.844	0.059
G3	1.728	0.032
G4	1.683	0.041
G7	1.597	0.064
RAM	1.462	0.035
Summed up	1.548	0.049

Algorithm 5 Classifier by association.

```

1 build classifier( $R, T$ )
   Input : The generated list of rules  $R$ , a set of
           transactions  $T$ 
   Output: A classifier by association
2  $R \leftarrow \text{sort}(R)$ 
3  $C \leftarrow \emptyset$ 
4 foreach rule in R do
5      $temp \leftarrow \emptyset$ 
6     foreach t in T do
7         if t satisfies antecedents of rule then
8              $temp \leftarrow temp \cup \{t.id\}$ 
9         if t satisfies consequents of rule then
10            mark rule as valid
11 if rule is marked as valid then
12      $C \leftarrow C \cup \{r\}$ 
13     Remove all  $t$  stored in  $temp$  from  $R$ 
14     Select default class for current  $T$ 
15     Compute the total number of errors of  $C$ 
16 Find the first rule in  $C$  with the lowest number of
    errors and remove the rest of the rules after it
17 Add a last rule that returns the default class after rule
    and append this to  $C$ 
18 return C

```

4 Results

4.1 Anomaly detection

The proposed AD method is able to detect anomalies online after a short period of warm-up. In our tests, this was as low as half an hour or just 30 samples. Table 3 shows the amount of detected anomalies for two different thresholds.

In order to get insight of the anomalies, these were clustered using HDBSCAN [17]. HDBSCAN is an extension of the density-based spatial clustering of applications with noise (DBSCAN) which allows clusters with different densities. The algorithm groups samples in high-density spaces and reports as anomalies samples in low-density spaces. In this way, anomalies that reoccur can be grouped together, and those that seldom occur are kept aside.

The anomalies are in most cases related to the magnitude of the main lobes. The magnitude of the acoustic signals is related to the force applied by the rotor; hence, deviations indicate that more or less force was required at a certain portion of the rotation. The discovered clusters were manually investigated and are summarized in Table 4. Figure 8 displays examples of the anomalies. The most common patterns discovered are:

Table 4 Validated anomalies and their diagnosis

Channel	%	Diagnosis	Fig.
G0	2.92	Less force required to punch through More force required to retract	a
	83.95	Earlier peak of the main lobe Additional peaks after main lobe	b
	13.13	Others	–
G3	31.36	Less force required to punch through Abnormal large peak afterwards	c
	30.80	Less force required to punch through Abnormal small peak afterwards	d
	37.84	Others	–
G4	87.98	Less force required to punch through	e
	12.01	Others	–
RAM	96.91	Less force required to punch through	f
	3.09	Others	–

G2 and G7 are not reported as no clusters were discovered. Visual examples of the anomalies are shown in Fig. 8

- *Decrease in magnitude of main lobe.* Appears in channels G0, G3, G4, and RAM. It is related to punches which required less force to go through.
- *Appearance of second lobe or reverberations.* Appears in channel G0. It occurs when the press required more force to return to the initial position.
- *Distortions before or after main lobe.* Occurs in channels G0, G3, and G4. Peaks or reverberations appear before or after the main lobe. These events need to be further investigated.

HDBSCAN was not capable of finding significant clusters for the anomalies in the channels G2 and G7.

In all channels, there was a fraction of anomalies which could not be clustered; we refer to them in Table 4 as others. These groups are composed of events that are extremely rare and therefore could not be clustered. Among them are events which do not have significance for PdM tasks such as disconnected sensors, noise, and logging errors; and events that may be informative such as punches requiring additional force, or other abnormal patterns. Figure 9 shows examples of patterns which were not clustered. If enough samples are labelled, then a supervised method can be trained to classify the less common anomalies.

4.2 Association rules

The mining of salient subsequences offers a considerable compression as it requires only $\leq 6.85\%$ of the data to store the most representative patterns. This representation

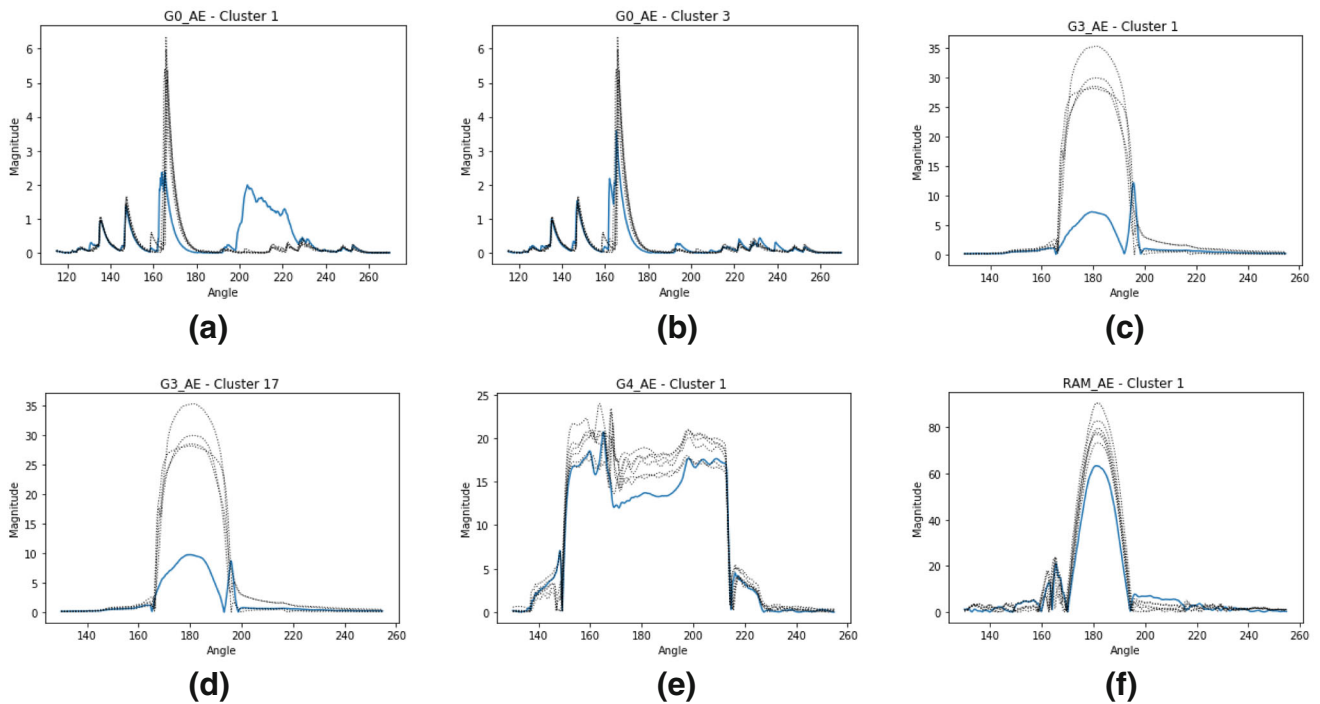


Fig. 8 Visualization of different anomaly types per channel. Each panel shows the centroid of the anomalous cluster (solid blue line) and as reference the most common patterns (dotted black line). **a** Anomaly in G0 with a smaller main peak and appearance of a secondary large peak. **b** Anomaly in G0 where main peak arises earlier. **c** Anomaly

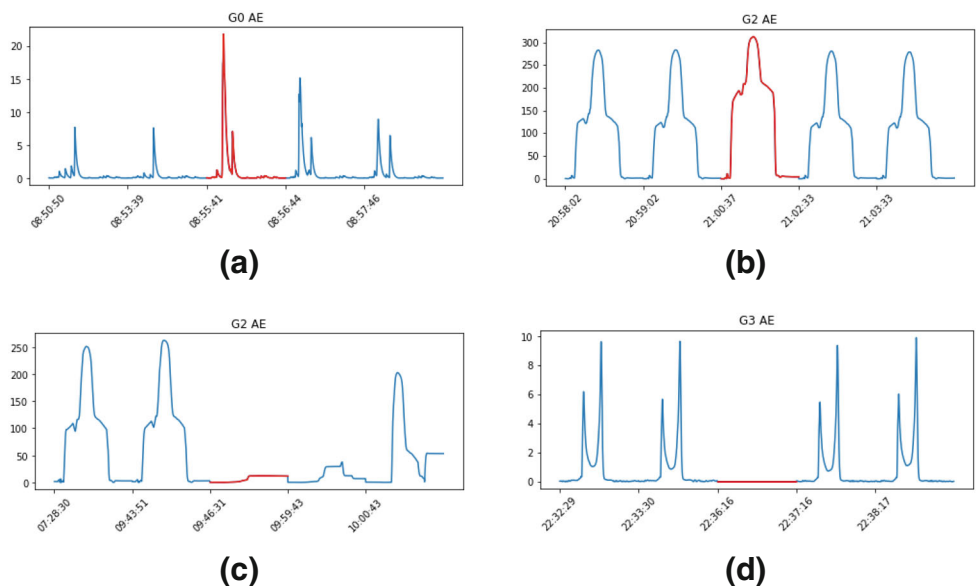
in G3 where the main lobe is smaller and a large peak appears afterwards. **d** Anomaly in G3 where the main lobe is smaller and a small peak appears afterwards. **e-f** Anomalies in G4 and RAM where the magnitude of the main lobe is lower

was then projected using PCA and preserving only 10 components, which are able to explain 94.11% of the variance. The compression percentage can be varied by tuning the number of trivial matches to ignore (n_t). However, the results are similar within a broad range of compression levels. Other aspects such as the number of PC

to preserve, the number of clusters generated, and the time window selected have a more significant effect.

Figure 10 shows the clustering using windows of 1 h and up to 80 clusters. From a visual inspection, it is evident that certain clusters are highly related to the events, but that some events may also happen unexpectedly.

Fig. 9 Examples of anomalies that were not clustered. **a, b** Punches requiring more force. **c, d** Possibly logging errors or sensors getting disconnected



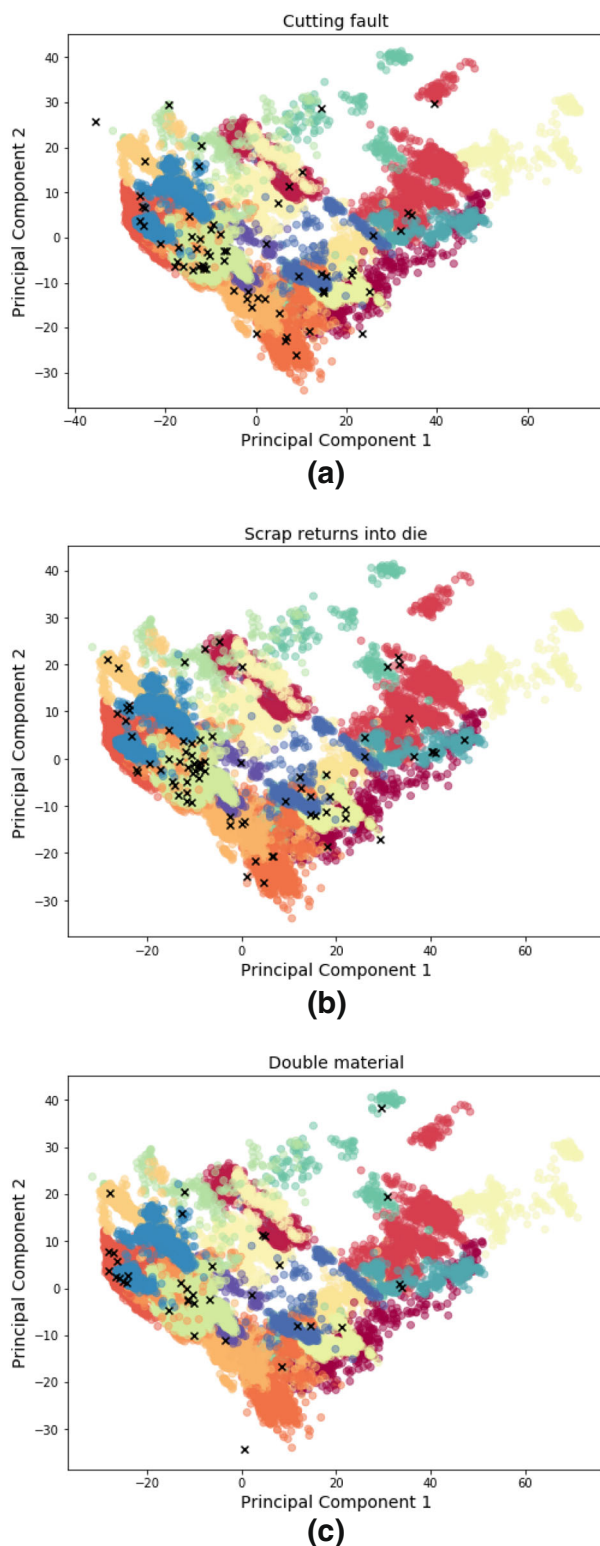


Fig. 10 Figures showing the discovered clusters using GMM and projected using the first two principal components. Each color corresponds to a different cluster. The “x” marks correspond to the last registered punch before the event occurred. **a** Cutting faults occur after a broken or a blunt punch. **b** Scrap returns into the die triggering a maintenance event. **c** Double material events occur when the position sensors detect a misalignment caused by double material being present

The rule mining technique was able to discover different sets of rules that trigger three out of the four events of interest: cutting faults, scrap returning into the die, and double material. No relevant rules were discovered for lubrication problems, which was an event with only six events over the year and therefore extremely rare. Table 5 shows the sets of rules discovered for a different number of clusters and time windows. The rules have been grouped according to the consequent they predict. These groups can be seen as sets of rules joined by an OR operator.

To compare the results, we use a baseline classifier which predicts the majority class in all cases, which corresponds to the healthy state. This way the model obtains a high micro F1-score (0.986–0.992) but does not predict the events. To take into account the class imbalance, we report the weighted F1-score. The best configuration is found at 60 clusters and windows of 1 h. It is important to notice that the time step has an important impact in the scoring of the classifiers as it reduces considerably the unbalance between healthy states and fault-related events. This increase in the score may be attributed more to the change in samples than more accurate rules. The best classifier is capable of detecting all events with a moderate F1-score (0.083 to 0.101) and a significant increase in the weighted score compared with the baseline. This is a relevant result considering that the fault-related events represent only 0.05% of the punches and 2.19% of the samples after windowing.

5 Discussion

The presented work brings time series analysis (TSA) and machine learning (ML) techniques to a real use case in the manufacturing industry and serves as a study of the viability of these techniques for PdM. The major lessons learned are presented below:

Integration in practice In order to create solutions that can be easily integrated into practice and cause minimum disruption, it is important to use methods that do not require extensive tuning or interaction from technicians. This was achieved by using the matrix profile technique that is domain agnostic and does not require tuning. In this case, the technique was effective enough to give results with only a short warm-up of few samples (30 samples, which is approximately 30 min). In the case of the event detection, the only step that requires careful tuning is selecting an adequate clustering algorithm. In some cases, the selected algorithm may require fine-tuning its parameters.

Data sampling This was one of the main challenges in the use case. We hypothesize that the low sampling

Table 5 Results for the rule mining

Model	Antecedent	Cons.	Recall	Precision	Events	F1-Score	F1-Score			
							Weighted	Weighted baseline	Micro	Micro baseline
20 clusters 30 min	[('17', '6')]	CF	0.027	0.286	74	0.049	0.062 (+0.060)	0.002	0.648 (-0.345)	0.992
	[('2', '18'), ('2', '12')]	DM	0.089	0.103	43	0.095				
	26 rules	H	0.499	0.986	13,645	0.663				
	['No rules']	SR	0.000	0.000	93	0.000				
20 clusters 60 min	[('2')]	SR	0.189	0.057	90	0.087	0.088 (+0.085)	0.003	0.628 (-0.358)	0.986
	[('2', '18'), ('2', '12')]	DM	0.116	0.156	41	0.133				
	24 rules	H	0.491	0.977	7012	0.653				
	['No rules']	CF	0.000	0.000	74	0.000				
40 clusters 30 min	[('17')]	SR	0.118	0.058	93	0.078	0.033 (+0.031)	0.002	0.647 (-0.346)	0.992
	[('7', '11')]	CF	0.027	0.400	74	0.051				
	48 rules	H	0.497	0.987	13,645	0.661				
	['No rules']	DM	0.000	0.000	43	0.000				
40 clusters 60 min	[('17')]	SR	0.111	0.087	90	0.098	0.024 (+0.021)	0.003	0.630 (-0.356)	0.986
	44 rules	H	0.497	0.974	7012	0.658				
	['No rules']	DM	0.000	0.000	41	0.000				
	['No rules']	CF	0.000	0.000	74	0.000				
60 clusters 30 min	[('33')]	CF	0.108	0.058	74	0.075	0.086 (+0.085)	0.002	0.643 (-0.350)	0.992
	[('52', '35'), ('35'), ('52')]	SR	0.118	0.044	93	0.064				
	[('56', '14'), ('8', '48'), ('12', '11')]	DM	0.170	0.071	43	0.101				

Table 5 (continued)

Model	Antecedent	Cons.	Recall	Precision	Events	F1-Score	F1-Score		
							Weighted	Weighted baseline	Micro
60 clusters 60 min	72 rules	H	0.492	0.988	13,645	0.657			
	[('52', '35'), ('35'), ('52')]	SR	0.111	0.066	90	0.083	0.097 (+0.094)	0.003	0.632 (-0.353) 0.986
	[('56', '14'), ('8', '48')]	DM	0.136	0.080	41	0.101			
	[('57', '50'), ('33')]	CF	0.132	0.076	74	0.097			
80 clusters 30 min	74 rules	H	0.495	0.977	7012	0.657			
	[('34'), ('24'), ('34', '24')]	SR	0.117	0.054	93	0.074	0.018 (+0.016)	0.002	0.652 (-0.340) 0.992
	78 rules	H	0.505	0.986	13645	0.668			
	['No rules']	DM	0.000	0.000	43	0.000			
80 clusters 60 min	72 rules	CF	0.000	0.000	74	0.000			
	[('46')]	DM	0.073	0.060	41	0.066	0.088 (+0.085)	0.003	0.624 (-0.362) 0.986
	[('54', '75'), ('67'), ('2'), ('34', '24'), ('34', '24'), ('24')]	SR	0.206	0.077	90	0.112			
	[('50', '9'), ('62', '77'), ('77')]	CF	0.158	0.074	74	0.100			

CF cutting fault, DM double material, SR scrap returns, H healthy state. The antecedents correspond to cluster IDs, which are the same in cases where number of clusters is the same but differs among the other models

rate can be one of the limiting factors causing the AD to not provide useful information for fault prediction. Currently, the sampling rate is 60 samples per hour. As the press can produce hundreds of punches within a minute, acoustic patterns related to events or faults might get easily missed. Additionally, a higher and uniform sampling rate would allow to treat the signal as continuous, opening opportunities for the use of other advanced ML techniques, such as recurrent neural networks (RNN) and long–short-term memory (LSTM).

Extra sources of information In the presented use case, the sensor information of the dies and the maintenance logs were used to assess the machine status. With the great availability of sensors nowadays, it is easy to keep track of all stages during the manufacturing process (such as room temperature, material properties, conditions of machines at later stages, and quality control, among others) that could be valuable for assessing other aspects, i.e., product quality and product defects, as well as complex interactions between manufacturing stages.

Expert knowledge Machine learning tries to learn the patterns from the data. However, experts have a lot of knowledge on the manufacturing process considered. Including this expert knowledge within the model design can provide great insight. For example, the AE patterns discovered in the anomalies and clusters could be further investigated and labelled, so that this information can also be included in the modelling. One step further is the step towards hybrid modelling approaches that combine data-driven models with physical and/or expert knowledge to further improve the models.

Concerning future work, we have identified the following opportunities:

- The presented AD is fully online, processing each sample as it arrives. In some scenarios, an immediate response is not required and therefore delayed predictions can be offered. In this case, the right MP can be considered which provides additional information during the nearest neighbor search. The delayed predictions would also allow the calculation of the arc-count meta-series, which has been proven to be a good estimator for state transition problems and segmentation [8].
- The rule mining approach can be easily extended with new sources of information as long as they can be represented as discrete events. For example, the material properties such as thickness, hardness, and temperature. At the same time, the rules approach can be used to predict other types of events, such as quality of the product or defective products.
- The classifier by association is a simple and efficient implementation. However, it can be extended with bootstrapping to create rules with higher accuracy and more robustness [16]. Currently, bootstrapping methods are hard to apply given the small amount of faults recorded. As more data are collected in the next years, it will be possible to perform the required partitions.

6 Conclusions

We present a general approach for AD and event prediction for a cold forming manufacturing line. The models are fast, and require few tuning of parameters and no expert knowledge regarding the physics of the components.

The anomalous events were manually analyzed and confirmed to be anomalous punches from the press. The event prediction is capable of detecting fault-related events that occur in less than 0.05% of the time with a micro F1-score of 0.632.

Most importantly, this work provides evidence on the potential use of AE signals for diagnostic purposes in the manufacturing industry.

Acknowledgments This work has been carried out in the framework of the Z-BRE4K project, which received funding from the European Union’s Horizon 2020 research and innovation programme under grant agreement no. 768869.

Funding This research received funding from the Flemish Government (AI Research Program).

Compliance with ethical standards

Conflict of interests The authors declare that they have no conflict of interest.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons licence, and indicate if changes were made. The images or other third party material in this article are included in the article’s Creative Commons licence, unless indicated otherwise in a credit line to the material. If material is not included in the article’s Creative Commons licence and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this licence, visit <http://creativecommons.org/licenses/by/4.0/>.

References

1. Ahmad S, Lavin A, Purdy S, Agha Z (2017) Unsupervised real-time anomaly detection for streaming data. *Neurocomputing* 262:134–147. <https://doi.org/10.1016/j.neucom.2017.04.070>

2. Ali SM, Hui K, Hee L, Leong MS (2018) Automated valve fault detection based on acoustic emission parameters and support vector machine. *Alex Eng J* 57(1):491–498. <https://doi.org/10.1016/J.AEJ.2016.12.010>
3. Blei DM, Jordan MI (2006) Variational inference for Dirichlet process mixtures. *Bayesian Anal* 1(1):121–143. <https://doi.org/10.1214/06-BA104>
4. Das AK, Suthar D, Leung CK (2019) Machine learning based crack mode classification from unlabeled acoustic emission waveform features. *Cem Concr Res* 121:42–57. <https://doi.org/10.1016/j.cemconres.2019.03.001>
5. Dykas B, Harris J (2017) Acoustic emission characteristics of a single cylinder diesel generator at various loads and with a failing injector. *Mech Syst Signal Process* 93:397–414. <https://doi.org/10.1016/j.ymsp.2017.01.049>
6. El-Ghamry MH, Reuben RL, Steel JA (2003) The development of automated pattern recognition and statistical feature isolation techniques for the diagnosis of reciprocating machinery faults using acoustic emission. *Mech Syst Signal Process* 17(4):805–823. <https://doi.org/10.1006/mssp.2002.1473>
7. Geng Z, Puhan D, Reddyhoff T (2019) Using acoustic emission to characterize friction and wear in dry sliding steel contacts. *Tribol Int* 134:394–407. <https://doi.org/10.1016/j.triboint.2019.02.014>
8. Gharghabi S, Ding Y, Yeh CM, Kamgar K, Ulanova L, Keogh E (2017) Matrix profile VIII: domain agnostic online semantic segmentation at superhuman performance levels. In: 2017 IEEE international conference on data mining (ICDM), pp 117–126
9. Gottwalt F, Chang E, Dillon T (2019) Corcorr: a feature selection method for multivariate correlation network anomaly detection techniques. *Comput Secur* 83:234–245. <https://doi.org/10.1016/j.cose.2019.02.008>
10. Han J, Pei J, Yin Y, Mao R (2004) Mining frequent patterns without candidate generation: a frequent-pattern tree approach. *Data Min Knowl Discov* 8(1):53–87. <https://doi.org/10.1023/B:DAMI.0000005258.31418.83>
11. Keogh E, Lin J (2005) Clustering of time-series subsequences is meaningless: implications for previous and future research. *Knowl Inf Syst* 8(2):154–177. <https://doi.org/10.1007/s10115-004-0172-7>
12. Kolyshkina I, Simoff S (2019) Interpretability of machine learning solutions in industrial decision engineering. In: Le TD, Ong KL, Zhao Y, Jin WH, Wong S, Liu L, Williams G (eds) *Data mining*. Springer, Singapore, pp 156–170
13. Kontaki M, Gounaris A, Papadopoulos AN, Tsihlias K, Manolopoulos Y (2011) Continuous monitoring of distance-based outliers over data streams. In: 2011 IEEE 27th international conference on data engineering. IEEE, pp 135–146
14. Li Z, Li J, Wang Y, Wang K (2019) A deep learning approach for anomaly detection based on SAE and LSTM in mechanical equipment. *Int J Adv Manuf Technol* 103(1-4):499–510. <https://doi.org/10.1007/s00170-019-03557-w>
15. Liu B, Hsu W, Ma Y (1998) Integrating classification and association rule mining. In: *Proceedings of the fourth international conference on knowledge discovery and data mining. KDD'98*. AAAI Press, pp 80–86
16. Mabu S, Gotoh S, Obayashi M, Kuremoto T (2016) A random-forests-based classifier using class association rules and its application to an intrusion detection system. *Artif Life Robot* 21(3):371–377. <https://doi.org/10.1007/s10015-016-0281-x>
17. McInnes L, Healy J, Astels S (2017) HdbSCAN: hierarchical density based clustering. *J Open Source Softw* 2. <https://doi.org/10.21105/joss.00205>
18. Naskos A, Kougka G, Toliopoulos T, Gounaris A, Vamvalis C, Caljouw D (2020) Event-based predictive maintenance on top of sensor data in a real Industry 4.0 case study. In: Cellier P, Driessens K (eds) *Machine learning and knowledge discovery in databases*. Springer, Cham, pp 345–356
19. Shanbhag VV, Rolfe BF, Griffin JM, Arunachalam N, Pereira MP (2019) Understanding galling wear initiation and progression using force and acoustic emissions sensors. *Wear* 436–437. <https://doi.org/10.1016/j.wear.2019.202991>
20. Varga M, Haas M, Schneidhofer C, Adam K (2019) Wear intensity evaluation in conveying systems – an acoustic emission and vibration measurement approach
21. Von Birgelen A, Buratti D, Mager J, Niggemann O (2018) Self-organizing maps for anomaly localization and predictive maintenance in Cyber-Physical production systems. In: *Procedia CIRP*, vol 72. Elsevier B.V., pp 480–485. <https://doi.org/10.1016/j.procir.2018.03.150>
22. Yeh C, Zhu Y, Ulanova L, Begum N, Ding Y, Dau HA, Silva DF, Mueen A, Keogh E (2016) Matrix profile I: all pairs similarity joins for time series: a unifying view that includes motifs, discords and shapelets. In: 2016 IEEE 16th international conference on data mining (ICDM), pp 1317–1322. [10.1109/ICDM.2016.0179](https://doi.org/10.1109/ICDM.2016.0179)
23. Yeh CM, Van Herle H, Keogh E (2016) Matrix profile III: the matrix profile allows visualization of salient subsequences in massive time series. In: 2016 IEEE 16th international conference on data mining (ICDM), pp 579–588
24. Yeh CM, Kavantzias N, Keogh E (2017) Matrix profile VI: Meaningful multidimensional motif discovery. In: 2017 IEEE international conference on data mining (ICDM), pp 565–574

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.