From raw data to information for a continuous supervision of machinery in dynamic industrial environments: a case study

Calabrese F.*, Ferrari E.*, Lelli G.*, Regattieri, A.*

* *Dipartimento di Ingegneria Industriale (DIN), University of Bologna, Viale Risorgimento 2, 40124 – Bologna – Italy (francesca.calabrese9@unibo.it, emilio.ferrari@unibo.it, giovanni.lelli2@unibo.it, alberto.regattieri@unibo.it)*

Abstract: Prognostic Health Management (PHM) is a recent approach for the realization of Predictive Maintenance. In literature, there are many papers dedicated to that topic, as well as to each of its part, i.e. signal processing, feature extraction, diagnostic and prognostic. However, when approaching to complex systems operating in real industrial contexts, there are several problems that make difficult its application. First, sensors positioned on the machinery couldn't have been thought for maintenance purposes. From PHM-application point of view, this could lead to a time-consuming data pre-processing activity, due to (1) a huge unnecessary amount of data, collected at high frequencies and (2) their intermittent collection during the machinery tests. Second, machinery often work under different operating conditions, that may depend on the kind of product or material that they process; they are often spread worldwide, so the same operating condition could be actually implemented in slightly different ways based on the geographic area the plant is installed in. Operating conditions may be unknown during the data collection and even if they are known for a specific machinery, they could change for another machinery, resulting in the impossibility to adopt a same feature, or set of features, and a supervised algorithm for diagnostic for the same machinery. In this paper, a methodology for data pre-processing, feature extraction and condition recognition is introduced through a discussion on a real case study. In particular, the data pre-processing takes into consideration both the quantity and the intermittent nature of collected data, by conducting sampling activities, detecting unstable conditions and setting them apart from subsequent classification; feature extraction and class recognition are conducted automatically, adaptively, and in real-time, so to always know the condition under which machinery is operating and ultimately to make easier the real-time anomaly detection and prognostics.

Keywords: Prognostic Health Management; data pre-processing; semi-supervised learning; real-time diagnostic

1. Introduction

As one of the main pillars of industry 4.0, Predictive Maintenance (PM) is attracting more and more attention from all the community of researchers and industries. The main advantage of such a strategy is the possibility to optimize the scheduling of maintenance interventions, not only based on historical data of critical components, but also on their actual health condition, so to exploit their whole life, until they actually do not have the ability to work properly. As a result, the economic advantages of a planned maintenance, e.g., the reduction of the costs related to a maintenance intervention, come also together with those related to a maximum exploitation of the tangible assets and to a better management of spare parts (Mobley, 2002). In other words, all maintenance activities could be positively influenced by the introduction of predictive strategies.

The fundamental requirement for the application of predictive maintenance is the installation of sensors on critical components of a certain machinery, through which data related to its health condition can be collected. Then, as one of the most important approaches discussed in literature suggests, named Prognostic Health Management (PHM), several Machine Learning (ML) algorithms can be used in order to (1) extract relevant information from raw data, (2) find the relationships between that information and the condition or life stage in which the component or the whole system is, (3) build a degradation model that allows to predict its Remaining Useful Life (RUL). These steps are usually referred as feature extraction or Health Indicator (HI) construction, Diagnostic or Health Stage division and Prognostic or RUL prediction (Lei *et al.*, 2018). Although in literature there are many papers about PHM and its application, most of them use data collected during lab tests, thus in a controlled environment. Controlled environments, here, means that the data analyst will be provided with all information related components or systems under analysis, such as the environmental conditions, the operating conditions of the system or the faults that have been simulated during the test. On the other hand, it also means that tests could be designed to train a specific analysis model, and therefore the collected data is far from reality. For these reasons, industries find big issues when approach to the implementation of PHM in their plants. In particular, very often, has to be performed on machinery provided with sensors designed for other purposes, e.g., process control. The data acquisition is not performed continuously, but only during machinery testing or according to technicians needs. As a result, the dataset could be unstructured, intermittent and unlabelled:

1. Unstructured means that data are collected and extracted in different ways, i.e., from different sources

and with different frequencies, which makes their processing a very hard task (Lu *et al.*, 2017);

2. Intermittent means that the available data do not allow to have a direct comprehension of what happened during the life of the monitored system, but just in some time slots, making difficult to correlate the values assumed by each variable (signal extracted from sensor) and physical phenomena they are describing;

3. Unlabelled means that data are not associated with the condition they refer, which makes difficult to distinguish different operating conditions and health conditions from fault conditions, leading this way to possibly erroneous conclusions (Calabrese *et al.*, 2018).

Besides those peculiarities of the datasets collected from the fields, there is also an important factor that characterizes industrial plants, that is its dynamic nature. In this context, to operate in dynamic environment means that even if at a certain time, all operating and fault conditions are known for all components and systems, they will not be stable for a long time. New product variants can be added anytime, which influence the machinery behaviour and settings. Furthermore, environmental conditions influence machinery behaviours, so that machinery operating in different seasons or in plants that are spread worldwide behave differently (Ye, Hong and Xie, 2013).

Given that, the immediate conclusion is that traditional supervised and batch PHM approaches cannot be directly applied by an industry that is moving its first steps towards predictive maintenance. As a consequence, other solutions should be found. In (Calabrese *et al.*, 2019), we have demonstrated as online and incremental learning can help achieving a real-time anomaly detection. Here, *anomaly* stands for a change from the current condition, which may correspond to either a new operating condition or a fault occurrence. In this paper, we tackle the problem of transformation of raw data into information that feed incremental and adaptive learning models. In particular, we focused on two main tasks:

1. The incremental transformation from raw data to information, so to always extract the best relevant features even when the operating condition is not known *a priori*

2. The improvement of the methodology presented in (Calabrese *et al.*, 2019), so that it does not only allow to detect anomalies and new conditions, but also to recognize that the system is entering a well-known condition and thus assign points to existing cluster, accordingly. This can be considered an automatic labelling of observations.

The application of incremental learning to both feature extraction and fault detection allow models to be "trained" on the available data only, and then applied in streaming, to the same machinery, or to other machinery operating in different industrial contexts. At the same time, a real-time labelling is performed, so that proper degradation models for RUL prediction can be selected and more accurate batch analysis can be done when enough data is available.

The remaining of this paper is structured as follows. In section 2, a brief literature review on feature extraction methods is provided, mainly focusing on dimensionality reduction methods. In addition, the theoretical background of incremental Principal Component Analysis (IPCA) is provided. In section 3, a brief overview on incremental anomaly detection and clustering is presented, and the built algorithm described. In section 4, an industrial case study is presented, which confirms the strength of such an approach. In addition, some of the characteristics that datasets should have for being processed in streaming and provide good results are also pointed out. Finally, conclusions and potential weaknesses arose from the application of the proposed approach will also be highlighted, which will be the focus of future research on this topic.

2. Incremental feature extraction

Very often, when tackling pattern recognition tasks, large datasets, with a huge number of variable (columns), need to be processed. In the industrial filed and in relation to the predictive maintenance activities, variables of a datasets are represented by signal collected from sensors installed on machinery. When the number of sensors is high, it is not trivial to understand correlations between variables and operating conditions. Dimensionality reduction techniques can help reducing the complexity of the problem by extracting relevant and non-redundant information, so to increase the accuracy of classification or prediction of ML algorithms.

Dimensionality reduction techniques can be classified into feature learning methods and feature selection methods, based on whether they change or not the original feature space (Tang, Alelyani and Liu, 2014). Here, we focus on feature learning, as it can be applied to datasets without label, thus in a unsupervised way. Feature learning is the process of projecting the original feature space into a new feature space with lower dimensionality.

One of the most adopted feature learning methods in the context of fault diagnosis is the Principal Component Analysis (PCA) (Zhu *et al.*, 2018), which results to be very effective. Given a dataset X, of dimension m, PCA aims to find a set of orthonormal basis vectors of dimension $p < m$, which are called Principal Components (PCs), that maximize the variance over the dataset when it is projected onto the subspace spanned by these PCs. Basically, if we have data points in a two-dimensional space and we want to project them in one-dimensional space, what PCA does is to find the direction of the vector and the position of the points on that vector, which is expressed by coefficients, such that the reconstruction or projection error is minimized (Fig.1). To this aim, the covariance matrix of the dataset is first computed, from which the eigenvalues and the eigenvectors can be extracted. The eigenvectors correspond to the PCs, while the eigenvalue of the corresponding eigenvectors represent the variance associated to that PC. The PCs are selected so that the cumulative variance described by them is greater than a certain percentage (usually, from 90 to 99%).

PCA, as well as other dimensionality reduction techniques, are usually applied to batch data. This means that, at the time of the analysis, the dataset is fully available. However, if operating or fault conditions vary over time and are not

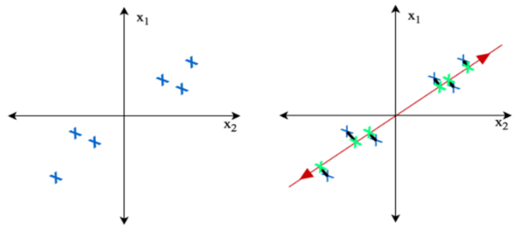

Fig. 1 Dimensionality reduction by PCA

known a priori, dimensionality reduction techniques may fail in extracting features that distinguish a new occurring condition. Rather, feature learning should be incremental and adaptive, so that when a new condition occurs, features that better characterize it, can be extracted.

When applied to streaming data, most important challenges are the following (Fujiwara *et al.*, 2020). First, the computation time needs to keep up with the data collection rate, meaning that the time for computing and updating the positions of data points has to be done before a new point is collected; second, when data is collected from different data sources, not all variables could be available at the same time, resulting in observations with varying dimensions. In (Diaz-Chito, Ferri and Hernández-Sabaté, 2018), a complete taxonomy of linear subspace incremental learning methods is provided and a comparison among methods has been done on different datasets. These methods perform well on streaming data, but still have some strict requirements on the kind of datasets that need to be processed.

In (Lippi and Ceccarelli, 2019), an exact incremental implementation of PCA is presented. As the authors state, *exacts* means that it provides the same results, i.e., the same PCs as in the batch version. In addition, it also contains an online data normalization, which is fundamental when variables assume very different values. Basically, the difference between the batch PCA and its incremental version formulated in that paper lies in the covariance matrix computation, which is recursive. In the incremental version, at a certain time $t + 1$, such a matrix is computed for $n + 1$ points, starting from the matrix computed for previous n points and including the new point $x + 1$. The authors also provided the algorithm for its implementation, which is available on the Mathworks website.

3. Incremental clustering

Clustering is a kind of unsupervised learning, used when no labelled data is available. Instead of predicting the class of a certain observation taking the target value into consideration at the prediction moment, unsupervised learning aims to find some structures in data and grouping them accordingly (Datta, A, Mavroidis, C, Hosek, 2007). Clustering algorithms group the data into clusters, so that that the distance, (the similarity), between points belonging to the same clusters is minimized (maximized), while the distance between different clusters is maximized (minimized). Since they do not need the target variable to be specified while learning, they are particularly suitable for streaming and online applications.

In general, clustering algorithms are classified into 5 main categories (Warren Liao, 2005): hierarchical methods, partitioning methods, density-based methods, grid-based methods and model-based methods. Although they work well, they generally require to know the number of clusters, whose shape is also defined, e.g., spherical. At the contrary, in streaming applications, the number of cluster could vary as new point is available and the shape of each cluster cannot be fixed, as the structure of the data is unknown.

In (Costa, Angelov and Guedes, 2015), a new paradigm has been developed for incremental clustering. It is based on the concepts of Recursive Density Estimation (RDE) and clouds, defined as free-shape clusters. Basically, it compares the global density, i.e., the density of all the points available up to a certain time stamp, with the local density, i.e., the density of all points since an anomaly had been detected up to the same time stamp, to decide whether the current point is still anomalous or can be considered normal and should create a new cloud.

In a previous work (Calabrese *et al.*, 2019), we applied anomaly detection algorithm (AAD) (Gu and Angelov, 2017) and the incremental clustering (ADP) (Gu, Angelov and Príncipe, 2018) together to recognize that a degradation was occurring and divide the whole life of the monitored components in a different life stages, in which the degradation was severer and severer.

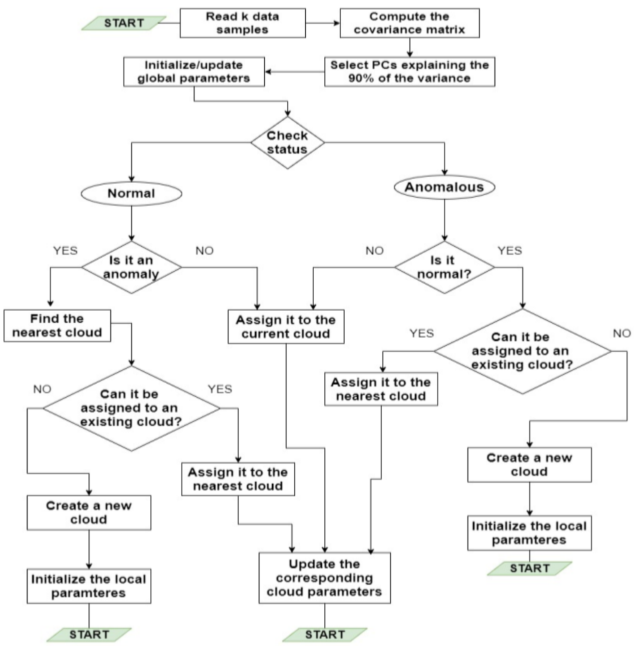

Fig. 2 The modified flow diagram of the proposed algorithm

Here, the algorithm has been improved to make it capable of assigning points to existing clusters, which corresponds to recognize a known condition. The flow diagram of the modified algorithm is depicted in Fig. 2. In addition, the incremental PCA has been included instead of energy feature extraction.

Basically, the algorithm, at each iteration, reads a sample, or a chunk made of k samples, computes the covariance matrix and selects the PCs that explain the 90% of the variance of the whole dataset. From here on forward, the vector of one observation, containing a number of variables equal to the selected PCs is considered as the feature which anomaly detection and clustering are based on. The first thing is now to check if the status of the system is set to normal or anomalous. In the first case, the global and local densities are compared in order to decide whether the current point is anomalous or normal. If normal, it is assigned to the current cluster, whose local parameters are updated. Otherwise, the nearest cluster represents a potential cluster for the current point. However, if the distance between the current point and the centre of this cluster is greater than the distance between all points read until now, the point creates a new cluster, whose local parameters are initialized. At the same way, when the status is detected as anomalous and the current point is still considered anomalous, then it is automatically assigned to the current cluster. Otherwise, the condition for assigning the point to the existing cloud or to a new cloud is checked again. That decision concludes the algorithm, which goes to the next iteration.

4. Case study

A case study of an automatic machinery is here presented. The focus will be on one of the components of machinery, which is subject to a very slow degradation. Since the machinery is designed to work with different kinds of materials, its operating conditions, which we call settings, may vary depending on the product to be produced. Accordingly, the degradation rate varies. Therefore, to compute the RUL based on the current degradation rate, it is fundamental to know at any point in time which values the machinery is set to. However, not all settings are known a priori. They are continuously modified, and each machine user can implement a setting depending not only by the product but also by the environmental conditions of the plant in which it is installed in.
The component is provided with 31 sensors, which measure the temperatures in different zones, as well as the percentage of use of thermo-resistors that are placed in each zone. In addition, the percentage of use and power of two electrical motors are also measured. All signals collected from the component are depicted in Fig. 3.
No one of these signals explain the setting changes by itself. Thus, feature extraction will be performed to reduce the dimensionality of the dataset and find the features that best explain the differences among the machinery settings. Since different settings are not known a priori, features that distinguish them cannot be known a priori as well. Thus, an

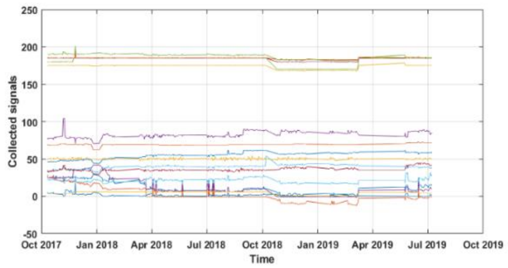

Fig. 3 Signals collected from the component under analysis

incremental algorithm for both feature extraction and clustering will be applied to the available datasets in order to

1. Extract relevant information (features) based on actual raw data
2. Recognize in which operating condition the machinery is operating

For feature extraction, we adopted the incremental PCA provided by (Lippi and Ceccarelli, 2019), since the dataset is made of a relatively small number of variable and data streams that potentially can grow quickly.

3.1 Dataset description

The available datasets were extracted from three sources. In the first data source, here named as S1, data were recorded once a day on a batch of 30 seconds after 30 minutes of machine functioning, so to avoid transients. Data of the second and the third data source, here named as S2 and S3, have been extracted from the machinery PLC at different frequencies, 1 Hz and 10 Hz, respectively. As summarized in Tab. 1, S1 covers a period of 2 years, while S2 and S3 cover just 8 and 9 months. In both cases, data are collected in an intermittent way, meaning that not all days are included in S1 and not all days and hours are included in S2 and S3.

Tab. 1 Datasets structure

Machinery	Source	Period of analysis
M1	S1	~ 2 years
	S2	8 months
	S3	9 months

Each data source contains a different number of variables. In particular, the data source S2 contains 27 variables, related to the temperatures and thermo-resistors of the component, while S3 contains variables related to the electric motors and the production rate. S1, finally, contains the variable of both datasets and also the setting values of machinery.

3.2 Methodology and results

First, a complete supervised and offline analysis is carried on, in order to verify that the PCs selected by the PCA are actually good features for classification. To this aim, we extract the different settings from S1. As shown in Fig. 4, 7 different settings were implemented in the period. In particular, two of them are just implemented for 1 day each. Then, the offline PCA is conducted on the variables of S1 (low frequency data source) to find the PCs that explain the most of variance contained in the dataset. As shown in Fig. 5, the 90% of the variance is provided by only 6 PCs. These PCs, were used to feed a linear SVM for classification, and it turned out that the accuracy of classification is 98.4%, against the value of 98.2% resulted from the application of the same classification model, without the prior application of PCA. Confusion matrices of SVM, from which accuracy

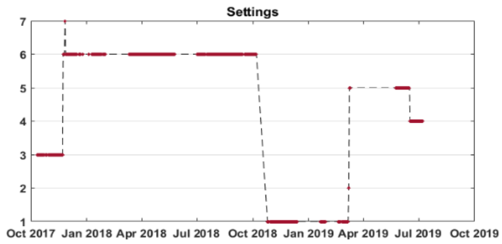

Fig. 4 Machinery settings implemented in the period of analysis

has been computed, are shown in Fig 6, with and without PCA respectively. The fact that the batch analysis with PCA outperforms the classification accuracy of the classification without PCA, justifies the effort for the incremental PCA. This is also justified by the fact that the PCs extracted incrementally are equal to those extracted offline, as shown in Fig 7.

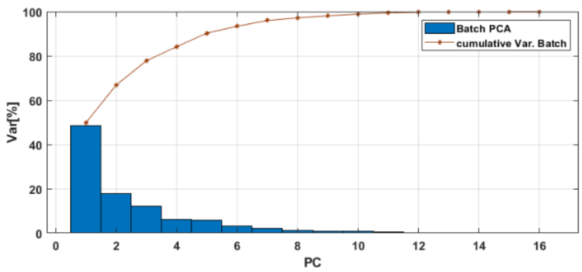

Fig. 5 Cumulative variance explained by PCs (batch)

Finally, the incremental PCA was embedded in the incremental clustering algorithm, in order to verify the goodness of the combination of these algorithms in the setting recognition. Results are shown in Fig. 8 and performances of the algorithm are summarized in Tab. 2, where the time stamp has been replaced by the number of the iteration, since data are not continuous in time.

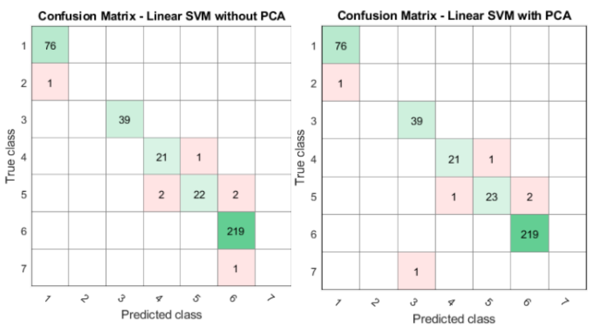

Fig. 6 Confusion matrices obtained by linear SVM without (on the left) and with (on the right) PCA

It resulted that the algorithm is able to recognize a change in the operating condition when there are enough points for each setting. Indeed, all the setting changes have been recognized, except for setting 7 and setting 2, that are represented by only one data point each. Note that the asterisks mean that since the previous cluster has not been recognized, the latency has been computed from the last recognized cluster. The high latency in both cases, is because when a point is considered anomalous, the algorithm does not evaluate the opportunity to generate a

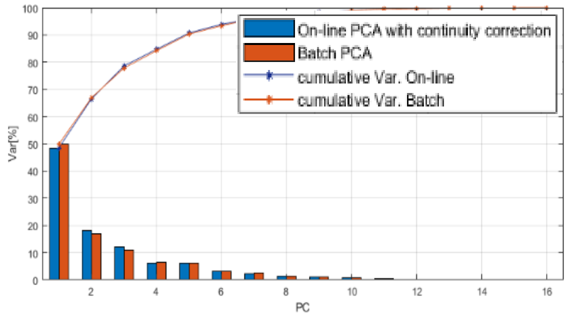

Fig. 7 Comparison on the cumulative variance explained by PCs with batch PCA and on-line PCA

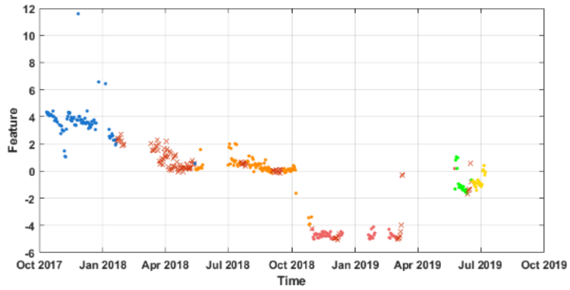

Fig. 8 Setting recognition by using incremental PCA and clustering

new cluster. This is only considered when the system comes back to the normal condition. For assessing the performance, we computed the accuracy as the number of correctly predicted points out of the total number of point. In this case, the accuracy is equal to the 70%. In addition, several false alarms have been generated. However, since they did not correspond to the creation of a cluster, they can be ignored.
Note that, the time to process a data point is much more lower than a second. Thus, with 'high latency', we actually mean that the algorithm takes few seconds for recognizing a change in the data structure.

Tab. 2 Clustering performance on S1

Setting change	True	Predicted	Latency
From 3 to 6	40	-	-
From 6 to 7	43	-	-
From 7 to 6	44	152*	112*
From 6 to 1	260	264	4
From 1 to 2	336	-	-
From 2 to 5	337	339*	3*
From 5 to 4	363	364	1

Since for the analysed component, high frequencies data sources S2 and S3 were also available, the same methodology was applied to them, in order to verify if a more high frequency could have led to a better accuracy. Note that, these dataset are not labelled. Thus, to evaluate the performance of the algorithm, only days for which the

setting values are known (from S1) are considered. S1 and S2 have been extracted between October 2017 and September 2018. As shown in Fig. 4, only 3 settings were implemented in that period (Setting number 3, 7, and 6). However, due to the intermittent collection, no data related to the setting 7 were included in S2 and S3. Therefore, only settings 3 and 6 were implemented.
Here, since the two sources were collected at different frequencies, many transients were included, and variables are split between the two sources, more data pre-processing was necessary, in order to join the two datasets and apply the proposed incremental procedure. First, to make the data sources homogeneous, the frequency of S3 (10 Hz) has been reduced to 1 Hz by computing the mean value over 10 samples. Then, the datasets were joined and transients eliminated by selecting samples for which the production rate was different to zero. In addition, during the online procedure, features (PCs) are extracted every 30 min (1800 points), so to feed the clustering algorithm with a more smoothed features. Results of PCA and clustering are shown in Fig. 9 and Fig. 10, respectively. It worth note that, in this case, the 90% of the variance is provided by 10 PCs. The algorithm is able to classify the observations into the 2 clusters with an accuracy of 96.75%. In particular, an anomaly is detected after 9 points from the real setting change, while the new cluster was created after 39 points.
As a result, we can conclude that dimensionality reduction techniques, as PCA, improve the classification even when the number of variables in not huge, in both batch and online analysis. When the data sampling frequency is low and few data is available for each condition, as in the case of settings 2 and 7, the anomaly detection algorithm, as well as the incremental clustering algorithm, fai~~ls~~ in recognizing them. However, this situation rarely happens in real industries that continuously collect the data from their machinery. Indeed, when data are collected at high frequencies, having one only data point per setting would mean that that a particular setting has been implemented for only 30 minutes (if the features are extracted every 30 minutes, as in the presented case). This result suggests that, even in the case of a slow degradation, if the condition recognition is required for adapting degradation modelling to different degradation rates, then recording of one observation per day (as in the case of data source S1) is not a good choice. Rather, a higher sampling frequency is necessary, so that more features representing the same operating condition can be extracted. In this way, the anomaly detection algorithm has enough "time" to recognize the change in the machinery behaviour and trigger the clustering algorithm. At the same time, a cluster corresponding to the same setting is well represented and can be distinguished by other clusters more easily. However, when data is collected at high frequencies, more PCs are needed to explain the variance of the data, which turns in a more required memory for the data storage.

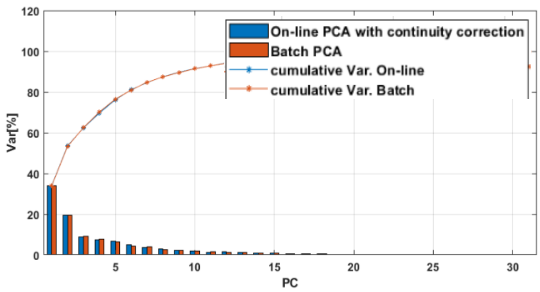

Fig. 9 Comparison on the cumulative variance explained by PCs with batch PCA and on-line PCA (on high frequencies data sources)

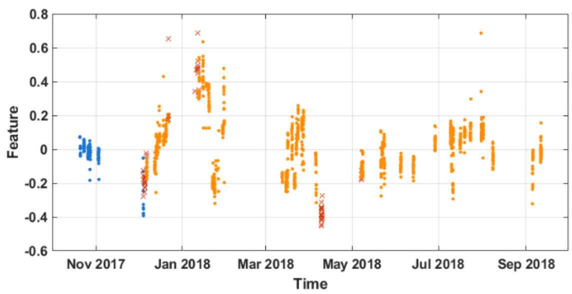

Fig. 10 Setting recognition by using incremental PCA and clustering (on hight frequencies data sources)

4. Conclusions and future research

In this paper, an unsupervised and streaming application of feature extraction and condition recognition is introduced, as fundamental parts for the implementation of predictive maintenance. In particular, the problem is tackled by an industrial point of view, highlighting which could be the goals and issues of an industry that is approaching to a prognostic program. Indeed, many industries have to deal with the following issues:

1. Datasets they collect are unstructured, intermittent and unlabelled, as in many cases they did not collect data for maintenance purposes;
2. When they analyse the data, they may do not know every possible operating conditions of machinery. In addition, settings may slightly different among each other for machinery installed in different plants.

These considerations led to resort to unsupervised and incremental learning, which can learn automatically and incrementally both relevant variables (features) and operating conditions (clusters).
To this aim, an incremental dimensionality reduction technique, IPCA, has been adopted for feature extraction, while the combination of AAD (for anomaly detection) and ADP (for clustering) has been used for the condition recognition. The application of these algorithms to a streaming dataset collected from an automatic machinery provided the following results. First, only few features out of 31 signals can be extracted. This represents an advantage for both the accuracy of clustering (classification) and the storage capacity required by the device eventually installed at the edge of the machinery. Second, based on those few features, the incremental clustering, starting from scratch, is able to correctly recognize different settings, when they are represented by a sufficient number of points. This represents an advantage for subsequent analysis. Indeed, a real-time condition recognition allow to update the corresponding degradation model and compute the RUL of the component accordingly. Finally, we have also seen that even in case of slow degradations, a higher sampling

frequency of data collection could lead to better results. In particular, in this case, we have demonstrated that with the data collection performed every 30 minutes, it is possible to achieve an accuracy for clustering of 96.75%.
In respect with the work presented in (Calabrese *et al.*, 2019), in which an online computation of the Health Indicator (HI) is performed, that refers to only one fault mode and was chosen based on an offline analysis, in the present work, a different set of features is extracted each time a change in the condition occurs (operating conditions or health condition). Thus, the improvement of the methodology means that, in this version, it is also able to recognize different conditions with no training.
Although this application demonstrates that the proposed approach achieves good results, still many efforts has to be done in order to: (1) improve the data collection procedure, so to make datasets as much structured as possible and fastening the data pre-processing step; (2) reduce the latency of the algorithm, so to recognize operating conditions represented by few points.

In addition, adaptive degradation models have also to be included into the algorithm for a complete prognostic program. The relationships between components of the same machinery and the influence that a fault in a component has on the other components, should also be taken into consideration. Further research will be dedicated to these topic.

References

Calabrese, F. *et al.* (2018) 'Components monitoring and intelligent diagnosis tools for Prognostic Health Management approach', in *Proceedings of the Summer School Francesco Turco*, pp. 142–148.

Calabrese, F. *et al.* (2019) 'Data-driven prognostics: from an offline and supervised analysis to an innovative, online and unsupervised methodology', *Proceedings of the Summer School Francesco Turco*, 1, pp. 74–80.

Costa, B. S. J., Angelov, P. P. and Guedes, L. A. (2015) 'Fully unsupervised fault detection and identification based on recursive density estimation and self-evolving cloud-based classifier', *Neurocomputing*. Elsevier B.V., 150(Part A), pp. 289–303. doi: 10.1016/j.neucom.2014.05.086.

Datta, A, Mavroidis, C, Hosek, M. (2007) 'A Role of Unsupervised Clustering for Intelligent Fault Diagnosis', *ASME International Mechanical Engineering Congress and Exposition, Proceedings*, 9, pp. 687–695. doi: 10.1115/IMECE2007-43492.

Diaz-Chito, K., Ferri, F. J. and Hernández-Sabaté, A. (2018) 'An overview of incremental feature extraction methods based on linear subspaces', *Knowledge-Based Systems*. Elsevier B.V., 145, pp. 1–14. doi: 10.1016/j.knosys.2018.01.020.

Fujiwara, T. *et al.* (2020) 'An Incremental Dimensionality Reduction Method for Visualizing Streaming Multidimensional Data', *IEEE Transactions on Visualization and Computer Graphics*, 26(1), pp. 418–428. doi: 10.1109/TVCG.2019.2934433.

Gu, X. and Angelov, P. (2017) 'Autonomous anomaly detection', in *IEEE Conference on Evolving and Adaptive Intelligent Systems*. Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/EAIS.2017.7954831.

Gu, X., Angelov, P. P. and Príncipe, J. C. (2018) 'A method for autonomous data partitioning', *Information Sciences*. Elsevier Inc., 460–461, pp. 65–82. doi: 10.1016/j.ins.2018.05.030.

Lei, Y. *et al.* (2018) 'Machinery health prognostics: A systematic review from data acquisition to RUL prediction', *Mechanical Systems and Signal Processing*. Elsevier Ltd, 104, pp. 799–834. doi: 10.1016/j.ymssp.2017.11.016.

Lippi, V. and Ceccarelli, G. (2019) 'Incremental principal component analysis: Exact implementation and continuity corrections', *ICINCO 2019 - Proceedings of the 16th International Conference on Informatics in Control, Automation and Robotics*, 1, pp. 473–480. doi: 10.5220/0007743604730480.

Lu, W. *et al.* (2017) 'Hybrid storage architecture and efficient MapReduce processing for unstructured data', *Parallel Computing*. Elsevier B.V., 69(2017), pp. 63–77. doi: 10.1016/j.parco.2017.08.008.

Mobley, R. K. (2002) 'Financial Implications and Cost Justification', *An Introduction to Predictive Maintenance*, pp. 23–42. doi: 10.1016/b978-075067531-4/50002-6.

Tang, J., Alelyani, S. and Liu, H. (2014) 'Feature Selection for Classification: A Review', *Data Classification: Algorithms and Applications*, pp. 37–64. doi: 10.1.1.409.5195.

Warren Liao, T. (2005) 'Clustering of time series data - A survey', *Pattern Recognition*. Elsevier Ltd, 38(11), pp. 1857–1874. doi: 10.1016/j.patcog.2005.01.025.

Ye, Z. S., Hong, Y. and Xie, Y. (2013) 'How do heterogeneities in operating environments affect field failure predictions and test planning?', *Annals of Applied Statistics*, 7(4), pp. 2249–2271. doi: 10.1214/13-AOAS666.

Zhu, Yonghua *et al.* (2018) 'Self-representation and PCA embedding for unsupervised feature selection', *World Wide Web*. Springer New York LLC, 21(6), pp. 1675–1688. doi: 10.1007/s11280-017-0497-2.