Data-driven prognostics: from an offline and supervised analysis to an innovative, online and unsupervised methodology

Calabrese F.*, Gamberi M*, Margelli S.*, Pilati F*, Regattieri A.*

* *Dipartimento di Ingegneria Industriale, Alma Mater Studiorum – University of Bologna, Viale del Risorgimento 2, 40136 - Bologna – Italy (francesca.calabrese9@unibo.it, mauro.gamberi@unibo.it, silvia.margelli@studio.unibo.it, francesco.pilati3@unibo.it, alberto.regattieri@unibo.it)*

Abstract: In industrial contexts, prognostics refers to the ability of predicting the Remaining Useful Life (RUL) of critical components and machines that are part of complex production systems. This is the rationale of a new maintenance strategy, named Predictive Maintenance (PM), whose ultimate goal is a near-zero downtimes situation. PM can be implemented through the so-called Prognostics Health Management (PHM) program, which is based on the monitoring of relevant parameters produced by the equipment, e.g., vibrations, in order to assess its health status. Based on the Condition Monitoring (CM) data and according to the PHM paradigm, the degradation process leading to an identified fault condition can be modelled and the RUL computed as the difference between the current time instant and the instant in which the monitored parameter is predicted to overcome a fixed Failure Threshold (FT). The supervised and offline approaches exploited to that purpose have important drawbacks limiting their application. In this paper, a data-driven methodology for the PHM implementation is introduced, which directly processes data in streaming and in an unsupervised way to detect anomalies, automatically find data partitions and compute the RUL as new data is available, after degradation models for clusters representing fault conditions have been identified. In this way, learning algorithms do not require large amount of historical data for training and never-seen incipient failures can be recognized during the machine functioning. The methodology has been applied to a rolling bearing vibration signal, whose degrading behavior until the fault has been simulated. Results show that the methodology is able to recognize a deviating behavior and anticipate the occurrence of the failure with sufficient time. The proposed case study also highlighted the necessity of a supervised and offline part for the FT and degradation model definition.

Keywords: predictive maintenance; Prognostics Health Management; data-driven methodology; online learning; unsupervised learning

1. Introduction

Prognostics Health Management (PHM) is a recently developed approach for the realization of predictive maintenance of complex production systems. Differently from Condition-Based Maintenance (CBM), that only has diagnostics capability, PHM also includes prognostics, that aims to anticipate the occurrence of a failure by predicting the Remaining Useful Life (RUL) of the component. In this way, maintenance interventions could be scheduled with sufficient time. The PHM takes advantage of several Machine Learning (ML) and Artificial Intelligence (AI) methods to get relevant information from the huge amount of raw signal collected from sensors and support the maintenance decisions (Jardine, Lin and Banjevic, 2006). In last years, many researchers have focused their efforts on improving ML methods for the PHM. In particular, supervised learning approaches have been largely investigated, as they allow to classify different fault classes and built reliable degradation models based on several training sets. Classification and regression models, as supervised tasks, learn rules from the training set and use them for future classification and prediction (Lolli *et al.*, 2018), (Alsina *et al.*, 2018). The term supervised refers to the consideration of a target variable, or label, during the learning process, that in the context of PHM corresponds to the fault condition (for classification) as well as the RUL (for regression) of the component/system. After data collection, the PHM suggests to conduct the following three main activities, (Shin *et al.*, 2018): signal processing or feature extraction, diagnostics, prognostics. The first step deals with the extraction of relevant characteristics from raw signals, named *features*, that are able to reveal the health status of the component and distinguish it from other health conditions; diagnostics deals with the identification of the relationships between the extracted features and the health status; prognostics deals with the degradation process modelling and ultimately with the prediction of the RUL. Traditionally, feature extraction is conducted by processing signals into the time domain, the frequency domain, and/or the time-frequency domain (Saufi *et al.*, 2017), in order to extract statistical features, such as the mean, variance, kurtosis, and other (Wang *et al.*, 2015), or energy features from raw signals. Often, PCA or deep learning, are also used as feature learning directly from raw signals, or dimensionality reduction and feature selection tools applied to a previously extracted feature set (Chalouli, Berrached and Denai, 2017). Diagnostics is mostly conducted by using classification models, such as Artificial Neural Networks (ANNs) (Gan, Wang and Zhu, 2016),(Alsina, Cabri and Regattieri, 2016), Support Vector Machines (SVMs) (Patel and Upadhyay, 2016), k-Nearest Neighbour (k-NN) or deep structures, like Deep Belief Networks (DBNs), Convolutional Neural Networks

(CNNs) and Recurrent Neural Network (RNNs). Most recently, to tackle the lack of training sets, unsupervised learning is also adopted for diagnostics purposes. Anomaly detection and clustering techniques are often combined in order to group observations into clusters, characterized by a high similarity among observations belonging to the same cluster and a low similarity among different clusters. In these approaches, anomalies or small clusters are representative of abnormal situations that may describe fault conditions (Wei *et al.*, 2017), (Rai and Upadhyay, 2017). While diagnostics is a post-failure analysis, prognostics aims to predict when the fault condition is going to happen. Therefore, instead of classification models, regression models are adopted to forecast the values that a properly selected Health Indicator (HI) is assuming in next feature and compute the RUL as the difference between the current time instant and the instant in which the HI is expected to exceed a pre-determined Failure Threshold (FT) (Lolli *et al.*, 2017). The RUL prediction is traditionally based on either physics or Condition Monitoring (CM) data models. Although physics-based models are more reliable, as they are based on the physic principles of components, they are difficult to build for complex systems. Hence, data-driven prognostics is adopted for modelling the evolution of the degradation for a specific fault behavior of a system. To this purpose, regression models, Bayesian methods as well as AI methods could be adopted (Shahraki, Yadav and Liao, 2017).

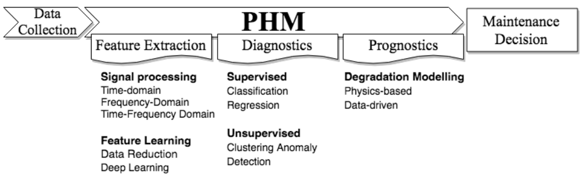

Figure 1: PHM for maintenance decision process

Typical applications of PHM rely on historical data sets that are stored and analysed in batch and in an *offline* mode (Calabrese, Regattieri, Bortolini, *et al.*, 2019). The RUL prediction is then implemented *online*, based on the results obtained in the *offline* step (Tobon-Mejia *et al.*, 2012), (Soualhi, Medjaher and Zerhouni, 2015). However, there are some issues that limit the application of this approach in real industrial contexts. First, historical information related to machinery is difficult to obtain since time-consuming accelerated tests should be conducted; in addition, even if several tests are carried out, it is not possible to know a priori all fault causes, and therefore simulate all possible fault conditions. Second, the massive data collected by sensors requires high storage capacity and often causes problems from the transmission point of view. Finally, the industrial environment is inherently dynamic and requires learning models to adapt as new data is available. The possibility to collect data in real-time is pushing towards streaming applications, where data are continuously collected. Streaming data are characterized by the concept drift, that refers to change over time of the data distribution, and the concept evolution, that refers to the possibility that new classes or clusters, representing new patterns, can occur at any point in time. Therefore, self-evolving models, able to handle streaming data (to deal with the concept drift and the concept evolution) for real-time anomaly detection, data clustering and classification are receiving great attention in the context of PHM.

In this paper, an unsupervised and *online* methodology able to detect anomalous behaviours, create data clusters for each identified condition, and predict the RUL is proposed. After the first time a fault condition is occurred, the proposed methodology is able to build a degradation model for each of the generated cluster during an *offline* procedure; then, the corresponding model is updated *online*, as new data is available, and the RUL is computed based on current data points. Therefore, the methodology (1) could be applied from "scratch", i.e., from the first read data sample, since it learns from new incoming points instead of historical datasets; (2) it could allow to store only the extracted features, reducing in this way the problems related to the data transmission and storage capacity; (3) since fault detection and RUL prediction can be carried out by light-weight algorithms, the *online* block could be implemented in edge devices, which would provide real-time information about the system behaviours; (4) it allows to collect the data while the machinery is functioning so that training sets can be accumulated for next accurate and reliable analysis with ML algorithms in a cloud/central server. Therefore, the proposed methodology could represent a starting solution towards the implementation of real-time predictive maintenance, having the potential to provide immediate feedbacks on the machine behaviours with minimal economical and time resources for obtaining historical data. The application of the methodology requires to select methods for extracting features, recognizing anomalies, clustering data as well as modelling the degradation behaviour from a large set of possibilities. The driver in this paper was the final application, that is represented by vibration signals of rolling bearings. These are non-stationary signals, often characterized by noise, to be analysed in streaming, with no prior cleansing activities. Therefore, time-frequency analysis, and in particular Empirical Mode Decomposition (EMD), has been chosen due to its ability to deal with non-stationary signals and its efficiency from the computational point of view. The Recursive Density Estimation (RDE) and the Autonomous Data Partitioning (ADP) have been chosen for anomaly detection and clustering, respectively. These methods are strictly related to each other and, to the best of our knowledge, they are the only ones that efficiently face the concept drift and concept evolution principles, considering a moving window of past data to compare with incoming data and creating clusters that can modify their shape as new data is available. Finally, the State-Space Model (SSM) has been chosen because of the characteristics of the simulated failure. Indeed, data collected are related to the degradation behaviour of rolling bearings, which can be described by an almost monotone HI. The SSM is a method that well fits data and can be efficiently updated when is inputted with monotone functions.

The remaining of the paper is organized as follows. Section 2 provides a theoretical background of the main algorithms adopted in the methodology, that are the EMD for feature extraction, the RDE for anomaly detection, ADP for data clustering, and the SSM for degradation modelling. Section

3 describes the necessary steps of the proposed methodology and provides a pseudo-code for its implementation that integrates all the described algorithms. Finally, section 4 provides an application of the proposed methodology to rolling bearings, whose degradation test data have been acquired using the PRONOSTIA platform (Nectoux *et al.*, 2012).

2. Theoretical background

In this section, the methods included in the proposed methodology are briefly described. In particular, the EMD is used for feature extraction in the time-frequency domain; RDE and ADP are described for anomaly detection and data clustering in streaming applications; the SSM is finally used for building the degradation model after a complete run-to-failure data set is obtained.

2.1. Empirical Mode Decomposition

Due to the non-stationary characteristics of the rotating machinery's vibration signals, like those produced by rolling bearings, processing signals into the time-frequency domain is more effective for feature extraction (Calabrese *et al.*, 2018). The Empirical Mode Decomposition (EMD) performs a time adaptive decomposition operation, according to which the signal is decomposed into a set of complete and almost orthogonal components, named as Intrinsic Mode Function (IMF). From the obtained IMF, it is possible to extract the energy of the signals in different frequency bands, that change when a fault condition occur (Bin *et al.*, 2012). The decomposition starts with the research of the maxima and the minima along the signal $x(t)$. Then, the maxima and minima are interpolated by means of two splines, $s_{max}(t)$ and, $s_{max}(t)$, that represent the boundaries of the signal. Finally, the mean function $m(t)$ is extracted and removed from the original signal, obtaining the new signal $x_1(t) = x(t) - m(t)$. This signal is an IMF if satisfies the following conditions (Rai and Upadhyay, 2017): (1) There must be an equality or a difference of at most one between the number of extrema and the number of zero crossings, and (2) the envelope defined by the local maxima and the envelope defined by the local minima must have zero average. The procedure is repeated until the obtained signal satisfies those conditions. When the conditions are satisfied, then the first IMF, $C_1(t)$, is obtained and subtracted from the original signal, obtaining the residual signal $r_1(t) = x(t) - C_1(t)$. The residual signal represents the input for the second IMF computation. The process stops when the residual signal is a constant or a monotonic function (Ricci and Pennacchi, 2011). From the obtained IMF, the energy features can be extracted as follows:

$$E_j = \sum_{k=1}^{n} C_i|(k)|^2 \tag{1}$$

where E_j is the energy of the jth subband for the IMF C_i and k represent the data sample.

2.2. Anomaly detection

The RDE is a concept used for fault detection in streaming data, which is based on the calculation of the density of a point, that determines how close a data sample is from the others at a certain time instant (Bezerra *et al.*, 2015). Data density is defined as the inverse of the sum of the total distances between all data points. If the Euclidean distance is considered, the density at a certain point is recursively computed as follows (Angelov, 2014):

$$D(x_k) = \frac{1}{1 + \|x_k - \mu_k\|^2 + \Sigma_k - \|\mu_k\|^2} \tag{2}$$

where

$$\mu_k = \frac{k-1}{k}\mu_{k-1} + \frac{1}{k}x_k, \qquad \mu_1 = x_1 \tag{3}$$

$$\Sigma_k = \frac{k-1}{k}\Sigma_{k-1} + \frac{1}{k}x_k, \qquad \Sigma_k = \|x_1\|^2 \tag{4}$$

μ_k and Σ_k are the global mean and the scalar product of the data sample x_k at the time instant k. Based on these parameters, the anomaly detection is realized by comparing this global density with a local density, computed as follows:

$$\mu_D = \left(\frac{ks-1}{ks}\mu_D + \frac{1}{ks}D(x_k)\right)(1 - \Delta_D) + D(x_k)\Delta_D \tag{5}$$

where

$$\Delta_D = |D(x_k) - D(x_{k-1})| \tag{6}$$

and ks is the number of feature vectors belonging to the same heath status. When an anomaly is detected and the status changes from one condition to another, ks is set to zero and the value of μ_D notably decrease. Hence, as data are collected and ks is incremented, μ_D will be greater than $D(x_k)$ and the first status can be re-established.

2.3. Autonomous Data Partitioning – ADP

The ADP algorithm has been introduced for the first time in (Gu, Angelov and Príncipe, 2018). In its recursive form, it partitions streaming data based on the concept of density, as introduced in section 2.2. Moreover, it uses the concept of cloud, instead of cluster, since the algorithms can generate arbitrary-shape cluster (Angelov and Yager, 2011). Each cloud is associated with two parameters: $c_{k,n}$, that is the prototype of cloud n at the iteration k; and $S_{k,n}$, that is the number of data points belonging to the cloud n at the iteration k. When the first feature vector is read, a first cloud is created and its parameters initialized as $c_{1,1} = x_1$ and $S_{1,1} = 1$. Then, for each feature vector $k > 1$, it computes the density at the prototypes of each cloud $D_k(c_{k,n})$. If the density at the current point is greater/lower than the density at any of the existing data cloud:

$$D(x_k) > \max_{i=1,\dots,nc}\left(D_k(c_{k-1,i})\right) \; OR \; D(x_k) < \min_{i=1,\dots,nc}\left(D_k(c_{k-1,i})\right) \tag{7}$$

than the current feature vector generates a new cloud. Otherwise, the cloud with the nearest prototype to the current feature vector is selected, C_{n^*}, and the condition expressed by Eq. 8 is checked: if the distance between the current feature vector and the nearest prototype is lower than the average distance between all the data samples, then the feature vector is assigned to the nearest cloud, whose parameters are updated by Eq. 9 and 10.

$$d(x_k, c_{k-1,n^*}) < \frac{\gamma_k}{2} \tag{8}$$

where $\gamma_k \sim \sqrt{\bar{d}_k} = \sqrt{2(\Sigma_k - \|\mu_k\|^2)}$

$$S_{k,n^*} = S_{k-1,n^*} + 1 \tag{9}$$

$$c_{k,n^*} = \frac{S_{k-1,n^*}}{S_{k,n^*}} c_{k-1,n^*} + \frac{1}{S_{k,n^*}} x_k \tag{10}$$

Otherwise, it creates a new cloud.

2.4. State-Space Models – SSM – for degradation modelling

The occurrence of a failure is often the result of a degradation process hidden into the collected parameters. However, the relationships between the collected parameters and the degradation of the system is often unknown. Therefore, State-Space Models (SSM) are often adopted for describing the stochastic process underlying the degrading progression of the failure (Jianmin, 2011). Given an unobservable state process $\{x_t\}_{t\geq 0}$ and an observation series $\{y_t\}_{t\geq 0}$, the SSM is completely specified by the initial state distribution $\pi(x_0)$ and the conditional probability density function $\pi(y_t|x_t)$ for $t \geq 1$

$$\begin{cases} \pi(y_t|x_{0:t}, y_{1:t-1}) = \pi(y_t|x_t) \\ \pi(x_t|x_{0:t-1}, y_{1:t-1}) = \pi(y_t|x_{t-1}) \end{cases} = \begin{cases} y_t = f(x_t) + v_t \\ x_t = g(x_{t-1}) + w_t \end{cases} \tag{11}$$

where x_t is the unobserved state of the system at time t, y_t is the observation at the time t, v_t and w_t are the process and measurements noises, respectively (Sun *et al.*, 2012). The first equation, named as observation equation reflects the relationship between the latent degradation condition and the indirect degradation indicator (collected parameter), while the second equation, named as state equation, reflects the evolution of the failure, i.e., the latent degradation condition. Given the SSM, the main task is to make an inference on the unobserved health state and predict the future state based on the collected parameters. Dynamic Bayesian methods are the most adopted for the state estimation of stochastic processes. Besides the specific degradation models, it is important that the observation y_t should be as monotonic as possible in time, so that it decreases/increases as the degradation increases. Therefore, for building an accurate degradation model, a proper Health Indicator (HI) should be selected, which can be either extracted directly from raw signals or computed from previously extracted features along the whole life of the component.

3. The proposed methodology

Based on the methods and algorithms described in previous section, an unsupervised and partially *online* methodology has been developed for fault detection, clustering and RUL prediction (Calabrese, Regattieri, Botti, *et al.*, 2019). The methodology can be divided into two main blocks: the *online* block takes as input streaming data, performs real-time processing, and gives as output a warning alarm when a fault condition is occurring. Results obtained from the ADP during the *online* step help determine the Failure Threshold (FT) over which the system is no more able to perform its activities and are used in the *offline* analysis to build the degradation model leading to the fault condition. The degradation model is finally used to predict, in the *online* block, the future behaviour of the system and ultimately its RUL, as the difference between the current time and the time in which the system is expected to exceed the failure threshold. The *online* procedure performs feature extraction, anomaly detection and data partitioning, until an anomaly is detected. Then, degradation modelling updating and RUL prediction are also included. The *offline* procedure is activated when an anomaly is detected. Thus, features computed until the anomaly is detected are collected, stored and used as input for the degradation modelling, which will be updated during the *online* procedure for HI values forecasting and RUL prediction. Note that, all the algorithms described in the previous sections have been integrated in a unique algorithm that, when detects an anomaly, also checks for cloud creation and updating procedure. The block diagram of the proposed methodology is shown in Fig. 2.

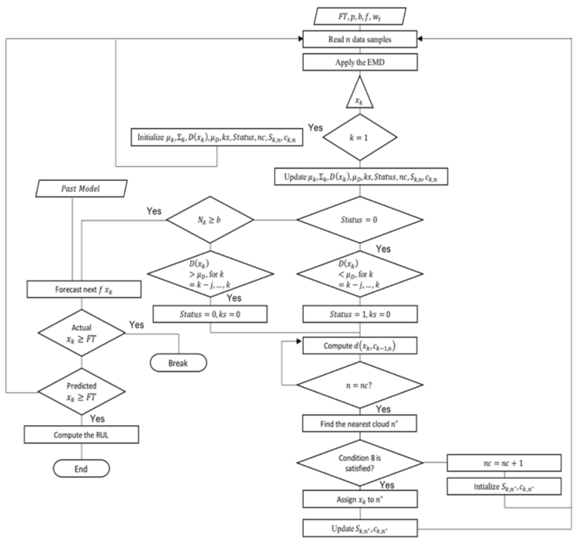

Figure 2: The proposed methodology

First, the time window, w_t, over which features are extracted has to be defined. For rotating systems, it could be, for example, equal to the cycle time of the machine. Then, for each signal segment, the EMD is computed in order to extract the energy feature x_k, that is taken as HI. For each feature x_k, computed at the k-th iteration, the mean value, the scalar product and the density are computed by Eq. (3), (4) and (2), respectively. In addition, parameters associated to the created cloud at the first iteration are updated by Eq. (9) and (10). Then, the condition (12) is checked (Costa, Angelov and Guedes, 2015)

$$D(x_k) < \mu_D \text{ for } k = k - j, \dots k \tag{12}$$

where j is an input parameter that represent the number of past values of the feature vector to compare with the current one. If the condition (12) is satisfied, then x_k is considered anomalous. At this point, the ADP is triggered.

In particular, if condition (8) is satisfied, then the current feature vector is assigned to the nearest cloud, whose parameters are updated by Eq. (9) and (10). Otherwise, it will generate a new cloud, whose parameters are initialized. The so-partitioned feature samples are therefore stored and used as HI for the SSM construction in the *offline* step.
After restoring the proper condition, the *online* procedure is re-activated. In particular, as shown in the diagram in Fig. 2, the inputs of the procedure are now the following: FT, w_t, the degradation model and the parameters related to the model updating, i.e., the number of HI values to forecast, f, the interval between two model updating, b, and the number of iterations over which the model has to be updated, p. Until the detection of an anomaly, the procedure remains the same as the first implementation. However, when an anomaly is detected and the feature has been assigned to a cloud, the updating of the degradation models is included into the procedure. Therefore, from the detection of the anomaly, f HI values are forecasted by the model based on the past p HI values, every b iterations. If the predicted HI values exceed the pre-determined FT, the RUL is predicted as the difference between the current time instant and the time in which the HI value is expected to reach the pre-determined FT. However, during the b iterations, it could happen that the actual HI value exceed the FT. In this case, the fault condition is occurred before the procedure was able to recognize it, causing an unexpected downtime. Therefore, the parameters b, f, and p should be properly selected, so that the best trade-off between computational requirements and accuracy of prediction is found. In addition, when an anomaly is detected, i.e., the status of the monitored component is changed, the condition 13 is checked (Costa, Angelov and Guedes, 2015)

$$D(x_k) > \mu_D \text{ for } k = k - j, \dots k, \tag{13}$$

That is, if the global density at a certain point is lower than the mean density of the points corresponding to the new status for the previous j points, then the current point is not considered anomalous anymore. This condition serves for restoring the “normal” condition after an anomaly has been detected, so that new anomalies can be recognized. In this way, the algorithm is able to handle situations in which a point is erroneously considered anomalous or anomalies occur due to some measurement errors. Note that, if training sets are available, this algorithm can be applied for the unknown fault conditions. Indeed, clouds related to the known fault classes could be initialized. When an anomaly is not assigned to any of the existing clouds, further analysis could be conducted to establish whether it corresponds to an actual fault condition and eventually trigger the *offline* analysis. In this way, next implementation of the procedure will also include the degradation model updating for the new discovered fault condition.

4. Case study

In this section, a case study for validating the proposed methodology is presented. The structure of PRONOSTIA platform and how the experiments have been carried out are both described in detail in (Nectoux *et al.*, 2012). Run-to-failure experiments have been conducted on rolling bearing in three different conditions. Data were collected from two acceleration sensors, attached horizontally and vertically to the tested bearing, with a sampling frequency of 25,600 Hz. Each 10 seconds, 2560 samples have been recorded. All bearings have different life durations, varying from 1 to more than 7 hours. In this case, only the first learning set of bearings in condition 1 is considered, whose horizontal acceleration during its life is depicted in Fig. 3.

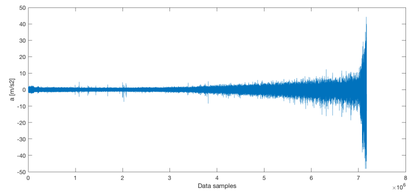

Figure 3: Run-to-failure data of Bearing1_1

The signal has been divided into several bins of 100 second ($w_t = 100$), each including 25,600 data samples. For each signal segment, the IMFs have been computed through the EMD method and the energy has been extracted by Eq. 1. Fig. 4 shows the IMFs corresponding to the last signal segment (on the left), and the energy feature of the signal (on the right).

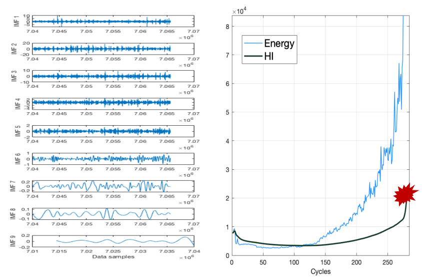

Figure 4: Energy and HI value extracted over the data set

Being the feature too oscillating, it is not suitable to be adopted as HI for degradation modelling. Therefore, the AAD has been applied to extract the mean (Eq. 3) of the energy computed recursively, which then has been adopted as HI (green line in Fig. 4). The SSM is built based on the HI values and the FT is set as the last value assumed by the HI. The fit to estimation data of the model is equal to the 99.24 %. Then, the *online* part of the methodology, is activated. The result is shown in Fig 5. a). It can be seen that the anomaly detection algorithm effectively recognizes when the component behaviour is deviating from the initial one (indicated by the red dots). Some anomalous points are also detected at the beginning and the RUL prediction is triggered.

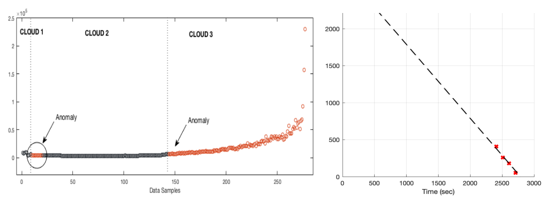

Figure 5: a) Anomaly detection. b) RUL prediction

However, when the status is considered "normal" again, (blue dots) the model updating and RUL prediction parts are not executed. This example justifies the presence in the algorithm of Condition 13. Indeed, the anomalies do not represent a failure nor an incipient fault. Rather, they could be present because of the time needed by the component to go to its regime functioning. This is also confirmed by the data partitioning algorithm, that recognizes three data clouds. The second cloud is created when the first anomaly is detected. However, even if the behaviour of the component is considered normal after a while, i.e., changes status, the data partitioning algorithm continues to assign data points at the current cloud, meaning that points are considered not deviating from the new behaviour. In contrast, when the second anomaly is detected, a third cloud is created. Until the end of the experiment, the behaviour is continued to be considered anomalous and data points assigned to the same cloud. This could mean that a fault condition is identified. From the first anomaly detected, the degradation model has been updated every $b = 10$ iterations. In particular, at each model updating, $f = 50$ HI values have been forecasted based on $p = 20$ previous HI values. At each iteration, the RUL has been predicted as the time difference between the current time and the time in which any of the forecasted HI values is expected to reach the fixed FT. In Fig. 5. b), the predicted RUL (red crosses) and the real RUL (dashed line) are depicted. The following table shows the values of the estimated RUL against the real RUL (expressed in minutes), computed after a certain number of cycles, with the corresponding error of prediction.

Table 1: RUL prediction results

Cycle	Estimated	Real	Error
242	68,3	63,3	+ 5
252	43,3	46.6	- 3,3
262	30	30	0
272	8,33	13,33	- 5

We can conclude that the first RUL estimation occurs after 6,7 hours (less more than one hour before the end of life), where the HI is predicted to be overcome the FT only 5 minutes later the real end of life. In the other cases, the prediction always anticipates the fault. However, the maximum loss of residual life is 5 minutes, which is acceptable.

5. Conclusions

In this paper, an unsupervised and partially online methodology for RUL prediction has been proposed and validated on a rolling bearing data set. First, the theoretical background of the models included in the methodology has been provided, i.e., EMD for feature extraction in the time-frequency domain, anomaly detection and ADP algorithms for fault detection and clustering, SSM for the degradation modelling. The methodology is then applied to a run-to-failure vibration data of a rolling bearing. The application shows the potential of the methodology in the context of predictive maintenance. First, the anomaly detection allows to determine in real-time if the behaviour of the component is changing from the correct functioning. In the meanwhile, the clustering algorithm allows to classify the whole life of the component in different health stages until the fault condition. The degradation model that is generated from this training set can be applied to different bearings, whose behaviour is similar, but not equal, to the previous one. Indeed, the model upgrading process considers both the original model and new incoming data. In addition, the mean computed recursively by the anomaly detection algorithm is always almost monotonic, which suggests that it could always be used as HI. These considerations make the methodology more generalizable and a practical solution for industrial contexts. Main limits regard the following aspects: first, some knowledge about the failure is still required to determine the FT and compute the HI; second, the possibility to have more fault classes is not considered; third, the algorithm should learn from false positives and adapt its input parameters as well as cluster structure based on this information Future research will be therefore focused on three points: how to reduce as much as possible the supervision of the phenomenon, how to automatically choose the correct degradation model from a set of models corresponding to different fault classes, how to let the algorithm learn from false positives.

References

Alsina, E. F. *et al.* (2018) 'On the use of machine learning methods to predict component reliability from data-driven industrial case studies', *The International Journal of Advanced Manufacturing Technology*, 94(5), pp. 2419–2433. doi: 10.1007/s00170-017-1039-x.

Alsina, E. F., Cabri, G. and Regattieri, A. (2016) 'A Neural Network Approach to Find The Cumulative Failure Distribution: Modeling and Experimental Evidence', *Quality and Reliability Engineering International*, 32(2), pp. 567–579. doi: 10.1002/qre.1773.

Angelov, P. (2014) 'Outside the Box : An Alternative Data Analytics Framework', 8, pp. 29–35. doi: 10.14313/JAMRIS.

Angelov, P. and Yager, R. (2011) 'Simplified fuzzy rule-based systems using non-parametric antecedents and relative data density', *IEEE SSCI 2011: Symposium Series on Computational Intelligence - EAIS 2011: 2011 IEEE Workshop on Evolving and Adaptive Intelligent Systems*, pp. 62–69. doi: 10.1109/EAIS.2011.5945926.

Bezerra, C. G. *et al.* (2015) 'A Comparative Study of Autonomous Learning Outlier Detection Methods Applied to Fault Detection', *2015 IEEE International Conference on Fuzzy Systems (FUZZ-IEEE)*. IEEE, pp. 1–7. doi: 10.1109/FUZZ-IEEE.2015.7337939.

Bin, G. F. *et al.* (2012) 'Early fault diagnosis of rotating machinery based on wavelet packets - Empirical mode decomposition feature extraction and neural network', *Mechanical Systems and Signal Processing*, 27(1), pp. 696–711. doi: 10.1016/j.ymssp.2011.08.002.

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., Regattieri, A., Bortolini, M., *et al.* (2019) 'PHM-based maintenance in complex systems: reference framework, competitive approaches, experimental evidences and future challenges', *Mechanical Systems and Signal Processing* (Submitted).

Calabrese, F., Regattieri, A., Botti, L., *et al.* (2019) 'Prognostic Health Management: an unsupervised and partially online methodology for fault detection and RUL prediction', *Mechanical Systems and Signal Processing* (Submitted).

Chalouli, M., Berrached, N. eddine and Denai, M. (2017) 'Intelligent Health Monitoring of Machine Bearings Based on Feature Extraction', *Journal of Failure Analysis and Prevention*. Springer US, 17(5), pp. 1053–1066. doi: 10.1007/s11668-017-0343-y.

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, 150(Part A), pp. 289–303. doi: 10.1016/j.neucom.2014.05.086.

Gan, M., Wang, C. and Zhu, C. (2016) 'Construction of hierarchical diagnosis network based on deep learning and its application in the fault pattern recognition of rolling element bearings', *Mechanical Systems and Signal Processing*. Elsevier, 72–73, pp. 92–104. doi: 10.1016/j.ymssp.2015.11.014.

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.

Jardine, A. K. S., Lin, D. and Banjevic, D. (2006) 'A review on machinery diagnostics and prognostics implementing condition-based maintenance', *Mechanical Systems and Signal Processing*, 20(7), pp. 1483–1510. doi: 10.1016/j.ymssp.2005.09.012.

Jianmin, Z. (2011) 'Remaining Useful Life Prediction Based on Nonlinear State Space Model', *2011 Prognostics and System Health Managment Confernece*. IEEE, pp. 1–5. doi: 10.1109/PHM.2011.5939528.

Lolli, F. *et al.* (2017) 'Single-hidden layer neural networks for forecasting intermittent demand', *International Journal of Production Economics*, 183, pp. 116–128. doi: https://doi.org/10.1016/j.ijpe.2016.10.021.

Lolli, F. *et al.* (2018) 'Machine learning for multi-criteria inventory classification applied to intermittent demand', *Production Planning & Control*. Taylor & Francis, 7287. doi: 10.1080/09537287.2018.1525506.

Nectoux, P. *et al.* (2012) 'PRONOSTIA: An experimental platform for bearings accelerated degradation tests', *EEE International Conference on Prognostics and Health Management, PHM'12*, (1–8).

Patel, J. P. and Upadhyay, S. H. (2016) 'Comparison between Artificial Neural Network and Support Vector Method for a Fault Diagnostics in Rolling Element Bearings', *Procedia Engineering*. The Author(s), 144, pp. 390–397. doi: 10.1016/j.proeng.2016.05.148.

Rai, A. and Upadhyay, S. H. (2017) 'Bearing performance degradation assessment based on a combination of empirical mode decomposition and k-medoids clustering', *Mechanical Systems and Signal Processing*. Elsevier Ltd, 93, pp. 16–29. doi: 10.1016/j.ymssp.2017.02.003.

Ricci, R. and Pennacchi, P. (2011) 'Diagnostics of gear faults based on EMD and automatic selection of intrinsic mode functions', *Mechanical Systems and Signal Processing*, 25(3), pp. 821–838. doi: 10.1016/j.ymssp.2010.10.002.

Saufi, M. S. R. M. *et al.* (2017) 'A review on signal processing techniques for bearing diagnostics', *International Journal of Mechanical Engineering and Technology*, 8(6), pp. 327–337.

Shahraki, A. F., Yadav, O. P. and Liao, H. (2017) 'A review on degradation modelling and its engineering applications', *International Journal of Performability Engineering*, 13(3), pp. 299–314. doi: 10.1555/10.23940/ijpe.xx.xx.xx.xxx.

Shin, I. *et al.* (2018) 'A Framework for Prognostics and Health Management Applications toward Smart Manufacturing Systems', *International Journal of Precision Engineering and Manufacturing-Green Technology*, 5(4), pp. 535–554. doi: 10.1007/s40684-018-0055-0.

Soualhi, A., Medjaher, K. and Zerhouni, N. (2015) 'Bearing Health Monitoring Based on Hilbert – Huang Transform, Support Vector Machine, and Regression', *IEEE Transactions on Instrumentation and Measurement*, 64(1), pp. 52–62. doi: 10.1109/TIM.2014.2330494.

Sun, J. *et al.* (2012) 'Application of a state space modeling technique to system prognostics based on a health index for condition-based maintenance', *Mechanical Systems and Signal Processing*. Elsevier, 28, pp. 585–596. doi: 10.1016/j.ymssp.2011.09.029.

Tobon-Mejia, D. A. *et al.* (2012) 'A data-driven failure prognostics method based on mixture of Gaussians hidden Markov models', *IEEE Transactions on reliability*. IEEE, 61(2), pp. 491–503. doi: 10.1109/TR.2012.2194177.

Wang, F. *et al.* (2015) 'A feature extraction method for fault classification of rolling bearing based on PCA', *Journal of Physics: Conference Series*, 628(1). doi: 10.1088/1742-6596/628/1/012079.

Wei, Z. *et al.* (2017) 'A novel intelligent method for bearing fault diagnosis based on affinity propagation clustering and adaptive feature selection', *Knowledge-Based Systems*. Elsevier B.V., 116, pp. 1–12. doi: 10.1016/j.knosys.2016.10.022.