Remaining Useful Life estimation based on discriminating shapelet extraction

Simon Malinowski, Brigitte Chebel-Morello, Noureddine Zerhouni

FEMTO-ST, ENSMM, Besançon, France

Abstract

In the Prognostics and Health Management (PHM) domain, estimating the remaining useful life (RUL) of critical machinery is a challenging task. Various research topics as data acquisition, fusion, diagnostics, prognostics and decision are involved in this domain. This paper presents an approach to estimate the Remaining Useful Life of equipment based on shapelet extraction. This approach makes use, in an offline step, of a history of run-to-failure data to extract discriminative rul-shapelets, i.e. patterns that are correlated with the RUL of the considered equipment. A library of rul-shapelets is hence extracted at this step. Then, in an online step, these rul-shapelets are compared to testing units and the ones that match these units are used to estimate their RULs. Therefore, the RUL estimation of a testing unit is based on patterns that have been selected for their high correlation with the RUL. This approach is hence different from classical similarity-based approaches that attempt to match complete testing units (or only late instants of testing units) with training ones to estimate the RUL. The performance of our approach is assessed with a case study on the remaining useful life estimation of turbofan engines and performance is compared with other similarity-based approaches.

1. Introduction

Remaining Useful Life estimation is one of the main tasks in the Prognostics and Health Management domain. The aim of any RUL estimation technique is to provide accurate prediction of the time after which an equipment will not be able to meet its operating requirements. RUL estimation is hence very important for industrial purposes as it can help in planning maintenance strategies, maximizing the useful operational life of equipment, reducing maintenance costs and avoiding breakdowns that might have critical impacts.

RUL estimation techniques in the literature are separated into three families : model-based, data-driven and hybrid approaches. Model-based approaches rely on building a physical model

Email addresses: `simon.malinowski@femto-st.fr` (Simon Malinowski), `brigitte.morello@femto-st.fr` (Brigitte Chebel-Morello), `noureddine.zerhouni@femto-st.fr` (Noureddine Zerhouni)

describing the degradation behavior of the equipment. For instance, stochastic filtering [1], particle filtering [2] have been used to model degradation in the case of fatigue crack growth. This kind of approaches is very accurate but requires the knowledge of the physical degradation of the system to be available, which is not always the case. In addition, in most of the cases, the equipment is composed of many components for which a different model needs to be defined. Furthermore, model-based approaches are closely related to the considered application. They hence lack genericity. Data-driven approaches make use of available run-to-failure data to extract relevant information based on a learning process. These approaches do not attempt to derive an analytical model from the data, but attempt to capture the relationship between sensory data and the degradation level of an equipment, in order to estimate its RUL. These approaches are usually easier to obtain and implement, but are often less accurate than model-based ones. They hence offer a trade-off between accuracy, complexity and applicability. Most of the data-driven approaches in the literature are based on machine learning, probabilistic or statistical tools. A good survey of machine learning and statistical techniques for prognostics can be found in [3]. Neural Networks have been widely considered to estimate the RUL [4–7]. Many other approaches based on neural networks techniques have been proposed. The authors of [8] have used echo state networks for RUL estimation, Javed et al. [9] developed a method combining wavelet based neural network and fuzzy clustering tools, deep belief networks are proposed in [10] for health state classification, and complex valued neural networks are used in [11]. Other machine learning tools have also been applied for RUL estimation, for instance support vector regression in [12]. Probabilistic and statistical tools have also been used for prognostics applications. Bayesian learning techniques [13], Wiener processes [14], copulas [15], Kalman filters [16], Hidden Markov models [17, 18] and autoregressive-moving-average models [19, 20] have also been considered in the literature. Hybrid approaches, for instance in [21–24], combine model-based and data-driven approaches. Similarly to model-based approaches, knowledge about the physical degradation of components is required for hybrid-ones, which lowers the applicability of such approaches.

[Figure 1 about here.]

In this paper, we focus on similarity-based approaches which are particular cases of data-driven ones. Fig. 1 depicts the general framework of such approaches. Sensory data (often multidimensional) is first processed : noise filtering, feature extraction, data fusion... Note that the kind of processing depends on the type of data. After these steps, data can sometimes be transformed into an 1-dimensional indicator, denoted *health indicator* (HI), that models the degradation of an equipment with a time-series (or trajectory). If the original sensory data already captures the health status evolution of the equipment, data can also be kept multidimensional. In Fig. 1, this step is depicted by the data processing box. After this step, data is formalized into instances. The kind of instances depend on the method : instances can represent complete HI trajectories, blocks of HI trajectories... Hence, in an offline phase, a library of instances is constructed from the training sensory data. In an online phase, the same processing and formalization operations are applied to testing sensory data (whose RUL is to be predicted) in order to obtain a new instance. This instance is then compared to the library of instances in the retrieval step and the most similar ones are selected in order to predict the RUL of the testing instance (RUL estimation box in Fig. 1).

The authors of [25] have won the 2008 PHM challenge with a similarity-based approach where instances were the HI's obtained after fusion of different sensor measurements and that relied on Euclidean distance between training and testing HI's to select most similar instances. In this method, the whole test trajectory is used to be matched to the library of training units. Another approach, in [26], only considers last parts of testing units HIs as instances for the matching, as last parts are more likely to be correlated with the degradation level. In this paper, we propose an alternative method for the instance formalization, retrieval and RUL estimation steps (gray boxes in Fig. 1), based on the extraction of relevant patterns from the training data.

Shapelets have been introduced in [27] and [28] for classification and early classification of time series. We extend here this notion and define a new kind of shapelets called rul-shapelets, that correspond to patterns correlated to the remaining useful life of an equipment. In this paper, in the formalization step, discriminative rul-shapelets are first extracted from a training set of trajectories representing run-to-failure data of an equipment. This step produces a library of rul-shapelets that will be used for RUL estimation of testing units. In the online phase, the testing instance is created and rul-shapelets from the library are searched in this testing instance. The rul-shapelets that are found in the testing instance provides an estimation about the RUL. Hence, the RUL estimation relies here on finding similar behaviors between patterns of the testing time series and the shapelets that have been extracted for their correlation with the remaining useful life. Hence, a major difference with other traditional approaches is that the RUL estimation is based on some parts of the test unit (and not the whole testing unit or only last instants), these parts being chosen because of their high correlation with the RUL.

This approach is compatible with any applications satisfying the following assumptions:

- Run-to-failure data is available
- Trajectories capturing the health status evolution can be obtained from sensory data
- Testing components are assumed to go through the same degradation process as training ones.

The rest of this paper is organized as follows. Section 2 gives an overview of research work on similarity-based RUL estimation technique and positions our proposed approach within this context. Section 3 describes how discriminative shapelets are extracted from a set of time series representing run-to-failure data. Section 4 explains how these shapelets are used to perform RUL estimation on testing units, and Section 5 evaluates the performance of the proposed approach on a case study.

2. Related work

In this section, related work about similarity-based approaches for RUL prediction is presented. We focus on explaining in details how the formalization, retrieval and RUL prediction steps are performed in the literature. Similarity-based approaches are relatively new for prognostics applications. Wang et al. [25] proposed a similarity measure between testing and training trajectories based on Euclidean distance. Instances are represented as 1-dimensional health indicator trajectories obtained by linear regression of the sensory data. In the online step, for a given training trajectory, the Euclidean distance between a testing trajectory and every sub-trajectories (shifted by different number of cycles) of the training one is computed and the minimum distance is retained together with the optimal shift. This procedure is repeated for every training trajectories. The best trajectories are selected according to minimum distances, up to a certain threshold. RUL is estimated from the RUL of the selected trajectories, using a weighted average. Xue et al. [29] proposed a fuzzy instance based approach for RUL prediction. The approach starts by building local fuzzy models for testing engines. The fuzzy model defines a cluster of peers in which each of these peers is a similar instance to the given engine with comparable operational characteristics. The final RUL estimate is obtained by aggregating the RULs of similar training instance via a similarity weighted sum. Zio et al. [30] proposed a similarity-based approach for prognostics using a fuzzy point-wise similarity defined for degradation trajectories. The distance score between two trajectories is based on a fuzzy membership function that maps the difference between trajectories elements into membership. Weights are inversely proportional to the distance score. RUL is then obtained as a weighted sum of RULs of similar instances. Ramasso et al. [26] proposed a method that jointly predicts observations (continuous states) and health states (discrete states) in order to estimate the remaining useful life of components. In this paper, the authors worked with multidimensional sensory data and the instances were chosen as multidimensional blocks of data. The instances hence are subsequences of the original sensory data. Instead of aggregating RULs of the most similar instances, the observation trajectories are aggregated to predict the future observations. Those observations are classified as health states and RUL is predicted as the time transition from the degrading to fault state. The retrieval phase of the algorithm is based on a Euclidean distance measure where only the last block of the testing trajectory is considered and compared to blocks of the library. In [31], Khelif et al. proposed a new similarity measure that aims at giving more weight to last observations (as they are more likely to be correlated with the degradation level), while using the total testing HI trajectory as instances.

In these works, the retrieval step is hence based on trying to match testing trajectories (or testing sensory data), or blocks of trajectories with training ones in order to estimate the RUL. This implies that every piece of trajectories in the training set carry relevant information about their correlation with the degradation level, which might not be the case. In this paper, we first rely on extracting, from the training set, patterns that are correlated with the remaining useful life of an equipment. These patterns are then matched to testing trajectories to estimate the RUL.

3. Shapelet extraction and selection

Let $\mathscr{T}$ be a training set composed of $|\mathscr{T}|$ time series, $T_1, \ldots, T_{|\mathscr{T}|}$. In this set, the time series may have different lengths. The length of the time series T_i is denoted by $l(T_i)$. With this notation, a time series T_i in the set $\mathscr{T}$ can be written $T_i = t^i_1, t^i_2, \ldots, t^i_{l(T_i)}$. For the sake of clarity, we assume here that the time series T_i are univariate, i.e. $t^i_j \in \mathbb{R}, \forall 1 \leq j \leq l(T_i)$. The feature extraction process described in this section can be extended to multivariate time series by applying it to every dimension of the time series.

As we focus in this paper on prognostics (estimation of the remaining useful life of equipment before failure), time series (or trajectories) in the set $\mathscr{T}$ represent a run-to-failure health indicator trajectory of an equipment. In this section, we aim at extracting, from the set $\mathscr{T}$ of time series, features that are correlated with the remaining length of a time series (time period between the instant when the feature is met and the failure of the equipment).

In the following, we consider features under the form of time series of small length (relatively to the average length of the time series in $\mathscr{T}$), that will be denoted *rul-shapelets*.

Definition 1: A rul-shapelet is defined by a tuple $f = (S, \delta, \mu)$, where $S = s_1, \dots, s_{l(S)}$ is a time series and δ is a distance threshold. μ represents the average value of the remaining length of a time series that is matched by f (cf. Definition 2).

Definition 2: A rul-shapelet $f = (S, \delta, \mu)$ is said to match a time series T if there exists a subsequence T' of T (of length $l(S)$) such that the Euclidean distance between S and T' is less than or equal to δ. In other words, f matches T if

$$\exists k \in [1, l(T) - l(S) + 1], \text{s. t.} \sqrt{\sum_{j=1}^{l(S)} (s_j - t_{k+j-1})^2} \leq \delta. \tag{1}$$

The match is hence defined at a time instant k. If more than one k satisfies (1), the instant of the match is defined as the one that leads to the minimal distance.

According to Definitions 1 and 2, when a time series T is matched by a rul-shapelet $f = (S, \delta, \mu)$ at a time instant k, we can estimate that the RUL of T (from time k) is μ, i.e. we estimate the length of the time series T to be equal to $k + \mu$.

In this section, we describe how to extract rul-shapelets from a set of time-series $\mathscr{T}$ and select the ones that convey sufficiently relevant information about the RUL.

3.1. Shapelet extraction

[Figure 2 about here.]

To extract a rul-shapelet f, we first need to extract a time series of small length (which represents the feature S of the rul-shapelet) from the set $\mathscr{T}$, and we then need to estimate the other three features associated with f : δ and μ. For that purpose, all the subsequences of lengths $l_1, \dots, l_N$ from the set $\mathscr{T}$ are first extracted and stored as it can be seen in Figure 2. The lengths l_i are parameters chosen by the user. Depending on the number of time series in $\mathscr{T}$ and their lengths, the number of subsequences extracted here can be very high. In order to keep a reasonable amount of shapelets, subsequences of length $l_i, \forall 1 \leq i \leq N$ can be quantized using the K-means algorithm into a smaller number of subsequences (only the centroids of the obtained clusters are kept). After this step, a set $\mathscr{F} = \{f_1, \dots, f_{|\mathscr{F}|}\}$ of shapelets is obtained. For all these shapelets, only the first feature S is known for the moment. In this case, a rul-shapelet f can also be written $f = (S, ?, ?)$. We explain in the following how to obtain the other two features that completely determine a rul-shapelet.

3.2. Shapelet selection

[Figure 3 about here.]

Definition 3: Let $f = (S, ?, ?)$ be a rul-shapelet and T a time series from $\mathscr{T}$. The best-match features (BMF) between f and T is defined as the pair (d, ρ), where :

$$\begin{cases} d = & \min\limits_{1 \leq j \leq l(T)-l(S)+1} \sqrt{\sum\limits_{i=1}^{l(S)} (s_i - t_{i+j-1})^2} \\ \rho = & l(T) - \arg \min\limits_{1 \leq j \leq l(T)-l(S)+1} \sqrt{\sum\limits_{i=1}^{l(S)} (s_i - t_{i+j-1})^2}. \end{cases} \tag{2}$$

In other words, d is the minimum Euclidean distance between S and a subsequence of T of the same length and ρ is the remaining time before the end of T when this minimum distance d is met. In the following, d will also be denoted by best-match distance and ρ by best-match RUL.

We can, according to Definition 3, compute the BMF between a rul-shapelet f and every time series of the set $\mathscr{T}$.

Definition 4: For a rul-shapelet $f = (S, ?, ?)$ and a set $\mathscr{T}$ of time series, we define the best-match features list between f and $\mathscr{T}$ as the list :

$$L_f = \langle bmf_1 = (d_1, \rho_1), \ldots, bmf_{|\mathscr{T}|} = (d_{|\mathscr{T}|}, \rho_{|\mathscr{T}|}) \rangle, \tag{3}$$

where $bmf_i, 1 \leq i \leq \mathscr{T}$ is the BMF between f and the i^{th} time series of $\mathscr{T}$. This list of pairs is then sorted so that the d_i's are in an increasing order ($d_1 \leq d_2 \leq \cdots \leq d_{\mathscr{T}}$).

On Figure 3 are shown the two first elements of a BMF list (bmf_1 and bmf_2) for a given example shapelet and a given training set. The left picture corresponds to the time series for which the best match distance d_1 is the smallest amongst the other time series. The shapelet is depicted by circles. The instant of the match is 68 here. This leads to a best-match rul ρ_1 of 120 (as the total length of T_1 is 188 here). On the right picture, we can see the second time series (in terms of best match distance). The best-match distance d_2 for this second time series is higher than d_1 but lower than the ones obtained for the other training time series of this example. The instant of the match is 83 for T_2, which leads to a best-match rul $\rho_2 = 112$.

We are now interested, for a rul-shapelet $f = (S, ?, ?)$, in finding the parameters δ and μ such that, when f matches a time series T (i.e. the best-match distance between f and T is lower than δ), we have a high confidence in estimating that the remaining time (from the instant of the match) before the end of the series T is *mu* in average. For that purpose, we use the best-match features list L_f between f and the set of time series $\mathscr{T}$. From this list, we first extract the list of best-match RULs: $R = \langle \rho_1, \ldots, \rho_{|\mathscr{T}|} \rangle$. This list R is normalized so that its average value equals 0 and its variance equals 1. We compute the index i ($2 \leq i \leq |\mathscr{T}|$) defined by:

$$i = \arg \min_{2 \leq j \leq |\mathscr{T}|} var[\rho_1, \ldots, \rho_j], \tag{4}$$

where *var* denotes the statistical variance. This equation also means that we are searching for the i first elements of R of minimum variance. In order to select only the most discriminative rul-shapelets, the shapelets yielding to a minimal partial variance (computed as in (4)) that is above a threshold τ can be discarded. When the values of R are normalized, the variance of the whole list is 1 (whatever the considered rul-shapelet). As τ gets close to 0, the shapelets become more discriminative.
Then, the parameters of the selected rul-shapelet f are computed as :

$$\begin{cases} \delta = d_i \\ \mu = (\rho_1 + \cdots + \rho_i)/i \end{cases} \tag{5}$$

where the values ρ_j correspond to the ones before normalization of R.

[Figure 4 about here.]

Example 1:

Figure 4 shows the steps described above to estimate the parameters of a rul-shapelet f. In this example, a training set of 100 training time series is used. Figure 4-(a) shows the values of the best-match RULs between f and the 100 trainingtime series. Note that these values are sorted according to the best match distances as explained above. The right part of Figure 4 shows the partial variances $var[\rho_1,\ldots,\rho_j], 2 \leq j \leq 100$. From these values, the index $i = 11$ is chosen according to Equation (4). The 11^{st} best-match distance (d_{11} in the best-match features list) is selected as the parameter δ and the 11 first values of the left image are selected (shown by triangles) to estimate μ according to Equation (5).

After these steps, the set $\mathscr{F} = \{f_1,\ldots,f_{|\mathscr{F}|}\}$ of rul-shapelets is filled, i.e. every f_i is defined with its three parameters.

4. Shapelet-based RUL estimation

In this section, we explain how we perform RUL estimation using a set of rul-shapelets $\mathscr{F}$ obtained as described in the previous section. The context is the following: we are monitoring the behavior of a component and our aim is to predict when this component is likely to face a failure, based on previous experiences. The monitoring of the component is modeled by a time series $U = u_1,\ldots,u_{l(U)}$. This time series is obtained as explained in the introduction after data processing and instance formalization (cf. Figure 1). No information is given about the last monitoring instant $l(U)$: it can be at an early stage of the life of the component, a late stage or any stage in between. The RUL estimation process described here aims at predicting the time difference between $l(U)$ and the failure of the testing equipment.

4.1. Extracting the rul-shapelets that match U

The first step of the RUL estimation described in this section is to find the rul-shapelets in $\mathscr{F}$ that match the testing time series U, and the time instant of the match when applicable. Let $f = (S, \delta, \mu)$ be a rul-shapelet of $\mathscr{F}$. The best-match distance between f and U is computed (i.e. the minimum Euclidean distance between S and a subsequence of U of length $l(S)$). If this best-match distance is lower than δ, then according to Definition 2, f matches U. The time instant idx_f of the match (i.e. the time index of the beginning of the subsequence of U that leads to the minimal distance) is stored together with f. If the best-match distance between f and U is greater than δ, then f is discarded. The same operation is repeated for all the rul-shapelets of $\mathscr{F}$, leading to a set $Match(U) = \{(f_1, idx_1), \ldots, (f_k, idx_k)\}$, where $f_i, 1 \leq i \leq k$ is a rul-shapelet and idx_i the time instant when f_i matches U. This set contains all the shapelets that match the testing time series U, and is used to estimate the RUL of U.

For some testing units, it might happen that no rul-shapelets match U. In that case, the set $Match(U)$ is empty and no RUL estimation can be made at that point. To prevent this from happening, the condition of the match (Definition 2) can be relaxed when $Match(U)$ is found to be empty, by adding a coefficient $\beta > 1$ to Equation (1) :

$$\exists k \in [1, l(T) - l(S) + 1], \text{s. t.} \sqrt{\sum_{j=1}^{l(S)} (s_j - t_{k+j-1})^2} \leq \beta \times \delta. \tag{6}$$

Relaxing this condition enables to perform a RUL estimation even when no match is found for a test instance U.

4.2. RUL estimation

Every rul-shapelet in $Match(U)$ conveys an information about the RUL of U given by the parameter μ of the rul-shapelet. Let $f_i = (S_i, \delta_i, \mu_i)$ be a rul-shapelet in $Match(U)$ and let idx_i be its matching time. According to this rul-shapelet, the estimated total length of U is $\mu_i + idx_i$. Taking into account all the information brought by the rul-shapelets of $Match(U)$, the estimated total length $\tilde{l}_U$ of U is given by

$$\tilde{l}_U = \frac{1}{k} \sum_{i=i}^{k} (idx_i + \mu_i) \tag{7}$$

Note that weights can also be inserted in Equation (7) to favor for instance rul-shapelets that match U at late time instants (as late time instants are closer to failures than early ones). We use in this paper a weight that corresponds to the ratio between the instant when the shapelet is matched and the length of the time series where it is matched. Ratios around 1 mean that the shapelet is matched close to the late instants.

5. Case study

To assess the performance of the approach proposed in this paper on a prognostics case study, we used turbofan engines data available on the NASA Prognostic Data Repository[1] [32]. We first describe the data, then give the performance metrics that we use, and finally present some results.

[1] http://ti.arc.nasa.gov/tech/dash/pcoe/prognostic-data-repository/

5.1. Turbofan data set

[Table 1 about here.]

The data available from [32] consists of multivariate sensory time series that have been gathered from turbofan engine dynamic simulation process. Run-to-failure data has been obtained using C-MAPSS, a simulation program for turbofan engines. Four different experimental setups have been conducted and have led to four different data sets. The settings of these experiments are given in Table 1. In this paper, we use two of these datasets : the #1 and the #4. The #1 is the easiest one as it considers only one type of fault and one operating mode, whereas dataset #4 is the most difficult one with two types of faults and six operating modes.

Run-to-failure data for the training units are composed of 21 sensor measurements at each cycle. The measurements start at a similar level of degradation, which is considered as healthy and stop when the equipment has reached a level that is considered not sufficient to meet its operating requirements. From these 21 measures, only 7 are kept here as they have been shown to be the most pertinent ones in [25]. These sensors used here are the number 2,3,4,7,11,12 and 15. This sensory data is corrupted by noise. A third-order polynomial curve is used to smooth the sensor values and keep only the trends of the time series. For illustration purposes, the 100 training time series of dataset #1 given by sensor 2 after smoothing are shown in Fig. 5-(a).

We have considered in this paper two different approaches for shapelet-based RUL estimation : the first one consists in working directly with the sensory data (the process explained in Sections 3 and 4 is done for each sensor) and the second one consists, as explained in the introduction, in computing a 1-dimensional trajectory called *health indicator* from the sensory data. For dataset #1, these two approaches can be considered and compared, whereas for dataset #4 only the second approach can be considered as the six operating conditions prevents from working directly with the sensory data. To obtain the HI trajectories, we have used linear regression. The linear regression maps the seven sensor values at each cycle into a real number between 0 and 1 (approximately). The coefficients of the linear regression are inferred using training samples. The training samples are chosen as follows : sensory data from the last 5 cycles of every training units are selected and assigned the value 0 (which means bad health state), and the sensory data from the first 5 cycles of the training units whose total life is above 240 cycles are selected and assigned the value 1 (healthy state). The coefficients of the regression are inferred from the training samples with a least square optimization. These coefficients are stored and will be used to construct HI trajectories of the testing units accordingly. For the dataset #4, six models are learned, one for each operating mode. The 100 training HI trajectories obtained for the dataset #1 are shown in Fig. 5-(b).

Note that, as explained in the introduction, our contribution in this paper concerns the retrieval and RUL estimation steps in a similarity-based RUL estimation scheme. Our approach makes use of HI trajectories in order to estimate the RUL of an equipment. In this paper, the HI trajectories are obtained as explained above from sensory data using linear regression. This method was used by Wang et al. [25] in their winning approach in the 2008 PHM challenge. They have then optimized techniques to obtain HI trajectories with more complex tools, as principal component analysis, kernel smoothing and the use of neural networks (cf. [33] for a complete study). As these approaches rely on heavy parameter settings and given that our contribution concerns the retrieval and RUL estimation steps, we used the approach presented in [25] and focus our results on showing the efficiency of our method given HI trajectories. Better overall performance might be obtained with optimized HI trajectories.

In the online phase, the RUL of the testing units are estimated using the testing sensory data. This data is incomplete: the measurements are stopped at a certain instant and the RUL from this instant needs to be predicted. Sensory data is smoothed and converted into a HI trajectory (if needed, depending on the considered approach) using the learned regression model(s), and the RUL estimation will be made as detailed in Section 4.

[Figure 5 about here.]

5.2. *Performance metrics*

We present and explain the different performance metrics that we use in this paper to assess the performance of the proposed RUL estimation method. These metrics are either taken from [34] or were defined by the PHM Challenge community. Let t be a time instant, $\hat{r}_t$ the RUL prediction made at t and r_t the real RUL at time t. Let also t_E represent the time at which the considered equipment reaches end of life ($t_E = t + r_t$).

5.2.1. *Prediction horizon*

Let $\alpha_h \in \mathbb{R}$ be a coefficient between 0 and 1 that represent the tolerance to errors. Let t_p be the first time instant when the absolute value of the difference between the predicted RUL and the real RUL is less than α_h times t_E. This time instant t_p is hence defined as follows:

$$t_p = \min\ t, s.t.\ r_t - \alpha_h \times t_E\ \leq\ \hat{r}_t\ \leq\ r_t + \alpha_h \times t_E. \tag{8}$$

As it can be seen in Fig. 6-(a), the prediction horizon PH is then defined as $PH = t_E - t_p$. The higher PH is, the better the prediction approach is. In addition, the prediction horizon should increase with α_h.

[Figure 6 about here.]

5.2.2. *Rate of acceptable predictions*

This metric evaluates the rate of predictions that fall in a cone-shaped region of acceptable prediction errors. This metric has two parameters : a time instant t_H from which the predictions are evaluated and α_r a coefficient in $[0,1]$ that represents the tolerance to errors (similarly to α_h above). The rate of acceptable prediction RAP is defined as:

$$RAP = \frac{Card(\{t \in \{t_H,\ldots,t_E\}, s.t.\ (1-\alpha_r) \times r_t\ \leq \hat{r}_t \leq (1+\alpha_r) \times r_t\})}{t_E - t_H + 1}, \tag{9}$$

where $Card()$ represent the cardinal of a set. RAP is a real number $\in [0,1]$. High values of RAP are associated with good estimation performance. Fig. 6-(b) illustrates how the RAP metric is computed. In this figure, t_H is set to 50 and α_r to 0.2. Estimations that fall in the cone-shaped region are depicted by circle points. The RAP metric is equal here to $19/43$. Note that estimations were done every 5 cycles in this example.

5.2.3. Relative accuracy

The relative accuracy (RA) evaluates the mean absolute percentage errors of RUL predictions for every time instant between t_H (set by the user) and t_E. It is formally defined as :

$$RA = 1 - Mean(\{\frac{|r_t - \hat{r}_t|}{\hat{r}_t}, t_H \leq t \leq t_E\}). \quad (10)$$

Values of *RA* close to 1 are associated with good estimation performance.

5.2.4. Prediction score

This metric was defined by the PHM community for the 2008 PHM challenge. It assigns a score *PS* to a prediction based on the difference between the prediction and the real RUL as follows.

$$PS = \begin{cases} e^{-(r_t - \hat{r}_t)/10} - 1 \text{ if } r_t - \hat{r}_t \leq 0 \\ e^{(r_t - \hat{r}_t)/13} - 1 \text{ if } r_t - \hat{r}_t > 0. \end{cases} \quad (11)$$

The lower the score PS is, the better the prediction is. Note that late predictions are more penalized than early ones, as late prediction can have harmful impact on the equipment.

In order to compute the first three metrics presented here (PH, RAP and RA), a complete run-to-failure testing HI trajectory is needed. Different subtrajectories are used from this complete trajectory in order to compute these three metrics. Hence, the testing data available in the Turbofan datasets are not compatible with these metrics. Indeed, in the testing data, trajectories are stopped at a given time and the rest of the trajectory is unknown (only the RUL is given for validation purposes). In the following, we present some results in terms of these measures (for the data set #4). To obtain these results, we had to split the training dataset into two parts : a real training part and a validation part in which the complete HI trajectories are used as testing ones. On the contrary, the fourth metric (prediction score) can be used with any kind of testing data. For this metric, we use the testing data available in the Turbofan dataset and we made RUL predictions from the available testing trajectory (after the last measurement of this trajectory).

5.3. Experimental settings and results

We present in this section the performance obtained with our proposed RUL estimation technique on two Turbofan datasets : the #1 and the #4. Three parameters influence the performance of the proposed technique : the lengths $l_1, \ldots, l_N$ of the extracted shapelets (Section 3.1), the number of centroids $n_1, \ldots, n_N$ used to quantize the shapelets (Section 3.1), and the discriminative threshold τ used to select the discriminative shapelets (Section 3.2). The lengths of the extracted shapelets needs to be chosen according to the distribution of lengths of the training units. Short lengths need to be taken in order to be able to cope with short testing units. In the considered application, the length of the training units vary from about 120 of 400. In our experiments, the lengths $10, 20, 30, 40$ and 50 have been chosen to extract the shapelets. The total number of extracted shapelets before quantization is here more than $16,000$ for each l_i. In the first experiment on dataset #1 (Section 5.3.1), we analyse the performance of our RUL estimation technique with varying values for the parameters $n_1, \ldots, n_5$ and τ. Conclusions about this are drawn in Section 5.3.1.

5.3.1. Dataset #1

In this first experiment, different values for the parameters $n_1, \ldots, n_5$ and τ are used. The values for $n_i, i \in 1, \ldots, 5$ are taken in the set $\{50, 100, 150, 200, 250, 300\}$. Note that we took $n_1 = \cdots = n_5$ in this experiment. The parameter τ is taken in the set $\{0.15, 0.25, 0.35, 0.45, 0.55\}$. In this experiment, the health indicator trajectories obtained from the sensory data are used. Table 2 sums up the performance obtained in terms of the average prediction score for the 100 testing units. We can observe that the performance generally improves when the value of the parameters n_i increases. Too few shapelets lead to bad estimation scores. When $n_i \geq 200$, the performance is somehow stabilized. We can also observe that when τ is very low ($\tau < 0.25$), the obtained scores are bad for every values of n_i. This is mainly because too many shapelets are discarded when τ is low. When $\tau \geq 0.25$, the performance also somewhat stabilizes. It can be seen from these experiments that when the parameters l_i and τ do not take very low values, the performance of the proposed RUL estimation technique is stable, and that the predictions are quite accurate. The best performance is obtained for $l_i = 300$ and $\tau = 0.45$ and is equal to 6.52. These parameters are used in the following.

Figure 7 represents the histograms of the prediction errors for three different approaches : (a) an estimation approach based on [25], (b) the proposed approach using the 7 dimensional time series and extracting rul-shapelets on each dimension and (c) the proposed approach using the health indicator time series obtained by linear regression. We can observe that the prediction errors are more concentrated around zero for the most right histogram. The prediction error range is $[-39, 62]$ for Fig. 7-(c). Authors of [26] have also applied their technique to this dataset. A similar error histogram is given, in which the prediction errors are more spread than the ones obtained with our approach. The range of prediction errors is about $[-80, 120]$ in [26].

[Table 2 about here.]

[Figure 7 about here.]

[Table 3 about here.]

Table 3 gives the performance of the proposed approach and compares it to a RUL estimation technique based on [25] in terms of the average score given in Equation (11). In the method proposed by Wang et al. [25], the RUL estimation depends on a parameter that determines the number of nearest neighbors kept. A training trajectory is kept if the Euclidean distance between this training trajectory and a testing one is less than γ times the minimum distance between the testing trajectory and all the training ones. We fixed this γ here to 3 as it gives the best results.

From this table, we can see that using the health indicator approach yields better results than working directly with the sensors. In addition, a better score is obtained using the shapelet-based approach than the one based on [25].

5.3.2. Dataset #4

We have also applied the shapelet-based RUL estimation approach to the dataset #4, which is the most difficult one. In order to compare our approach with the optimized approach of [33], in a first experiment we have split the training dataset into two parts, as explained above : 150

instances are kept for training and 99 for validation purposes. This splitting enables us to compute the three metrics PH, RAP and RA and compare the performance with the ones given in [33] for the same dataset and same configuration. The performance in terms of the metrics PH, RAP and RA of our approach is given in Table 4. In this table, we also report the best results obtained by Wang on a testing set of 99 trajectories (cf. results in the thesis [33]). The parameters for metric evaluation have been chosen according to [33], i.e. $t_H = 80$, $\alpha_h = 0.2$ and $\alpha_r = 0.2$. The values reported in this table are the median values obtained for the 99 testing instances. It can be seen that the performance obtained by our approach is very close to the ones obtained in [33]. Note that, as explained above, the modeling of the HI trajectory in [33] has been optimized with many different tools (PCA, Kernel smoothing and neural networks), whereas we have considered here a simple modeling with linear regression.

[Table 4 about here.]

In a last experiment, we have also considered the real testing set of dataset #4. In this testing set, 248 RULs have to be predicted from incomplete run-to-failure data. Table 3 reports the performance of the proposed approach and compares it to a RUL estimation technique based on [25] in terms of the average score given in Equation (11). The average score obtained by our approach is significantly better than the one obtained from a RUL estimation based on [25].

[Table 5 about here.]

6. Conclusion

We have proposed in this paper a RUL estimation technique based on shapelet extraction. The shapelet extraction process aims at selecting, from a training set of run-to-failure data, patterns (under the form of small time series) that are correlated with the remaining useful life of an equipment. These extracted patterns convey each an information about the RUL of the equipment from the instant when they are found. They are then used in an online step to estimate the RUL of testing units (units for which the remaining useful life is not known). Therefore, the proposed RUL estimation of a testing unit is based on patterns that have been selected for their high correlation with the RUL. This approach is a similarity-based approach : it makes use of available run-to-failure data (in which the health state evolution is captured) to estimate the RUL of equipment. It is hence compatible with techniques aiming at constructing health indicators from sensory data. We assessed the performance of the proposed RUL estimation technique on Turbofan datasets. The RUL estimation performance is shown to be effective and competitive compared to other similarity-based approaches. The proposed method is more adapted to applications where many training instances are available, in order to select relevant shapelets and to be able to provide an estimation of the RUL even at early life stages. Extending this method to limited training sets will focus our attention in a near future. This paper also opens up other possibilities of using shapelets in the field of Prognostics and Health Management. We are planning to extend the use of shapelets for diagnostic purposes by linking the apparition of shapelets with different kind of faults, or for predicting the future evolution of the health status of components.

Acknowledgment

The authors would like to acknowledge the European Regional Development Fund (FEDER) who funded this work as a part of a project named ALTIDE: Aid to Lifecycle Traceability for Intelligently Developed Equipment.

References

[1] E. Mytyri, U. Pulkkinen, and K. Simola, "Application of stochastic filtering for lifetime prediction," *Reliability Engineering & System Safety*, vol. 91, no. 2, pp. 200 – 208, 2006. Selected Papers Presented at {QUALITA} 2003.

[2] F. Cadini, E. Zio, and D. Avram, "Model-based monte carlo state estimation for condition-based component replacement," *Reliability Engineering & System Safety*, vol. 94, no. 3, pp. 752 – 758, 2009.

[3] J. Sikorska, M. Hodkiewicz, and L. Ma, "Prognostic modelling options for remaining useful life estimation by industry," *Mechanical Systems and Signal Processing*, vol. 25, no. 5, pp. 1803 – 1836, 2011.

[4] N. Gebraeel, M. Lawley, R. Liu, and V. Parmeshwaran, "Residual life predictions from vibration-based degradation signals: a neural network approach," *Industrial Electronics, IEEE Transactions on*, vol. 51, no. 3, pp. 694–700, 2004.

[5] R. Huang, L. Xi, X. Li, C. R. Liu, H. Qiu, and J. Lee, "Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods," *Mechanical Systems and Signal Processing*, vol. 21, no. 1, pp. 193 – 207, 2007.

[6] A. P. Vassilopoulos, E. F. Georgopoulos, and V. Dionysopoulos, "Artificial neural networks in spectrum fatigue life prediction of composite materials," *International Journal of Fatigue*, vol. 29, no. 1, pp. 20 – 29, 2007.

[7] F. Heimes, "Recurrent neural networks for remaining useful life estimation," in *Prognostics and Health Management, 2008. PHM 2008. International Conference on*, pp. 1–6, Oct 2008.

[8] Y. Peng, H. Wang, J. Wang, D. Liu, and X. Peng, "A modified echo state network based remaining useful life estimation approach," in *Prognostics and Health Management (PHM), 2012 IEEE Conference on*, pp. 1–7, June 2012.

[9] K. Javed, R. Gouriveau, and N. Zerhouni, "Novel failure prognostics approach with dynamic thresholds for machine degradation," in *Industrial Electronics Society, IECON 2013 - 39th Annual Conference of the IEEE*, pp. 4404–4409, 2013.

[10] P. Tamilselvan and P. Wang, "Failure diagnosis using deep belief learning based health state classification," *Reliability Engineering & System Safety*, vol. 115, no. 0, pp. 124 – 135, 2013.

[11] O. Fink, E. Zio, and U. Weidmann, "Predicting component reliability and level of degradation with complex-valued neural networks," *Reliability Engineering & System Safety*, vol. 121, no. 0, pp. 198 – 206, 2014.

[12] T. Loutas, D. Roulias, and G. Georgoulas, "Remaining useful life estimation in rolling bearings utilizing data-driven probabilistic e-support vectors regression," *Reliability, IEEE Transactions on*, vol. 62, pp. 821–832, Dec 2013.

[13] P. Wang, B. D. Youn, and C. Hu, "A generic probabilistic framework for structural health prognostics and uncertainty management," *Mechanical Systems and Signal Processing*, vol. 28, no. 0, pp. 622 – 637, 2012. Interdisciplinary and Integration Aspects in Structural Health Monitoring.

[14] K. L. Son, M. Fouladirad, A. Barros, E. Levrat, and B. Iung, "Remaining useful life estimation based on stochastic deterioration models: A comparative study," *Reliability Engineering & System Safety*, vol. 112, no. 0, pp. 165 – 175, 2013.

[15] Z. Xi, R. Jing, P. Wang, and C. Hu, "A copula-based sampling method for data-driven prognostics and health management," in *Prognostics and Health Management (PHM), 2013 IEEE Conference on*, pp. 1–10, June 2013.

[16] P. Baraldi, F. Mangili, and E. Zio, "A kalman filter-based ensemble approach with application to turbine creep prognostics," *Reliability, IEEE Transactions on*, vol. 61, pp. 966–977, Dec 2012.

[17] K. Medjaher, D. Tobon-Mejia, and N. Zerhouni, "Remaining useful life estimation of critical components with application to bearings," *Reliability, IEEE Transactions on*, vol. 61, pp. 292–302, June 2012.

[18] R. Moghaddass and M. J. Zuo, "An integrated framework for online diagnostic and prognostic health monitoring using a multistate deterioration process," *Reliability Engineering & System Safety*, vol. 124, no. 0, pp. 92 – 104, 2014.

[19] W. Wu, J. Hu, and J. Zhang, "Prognostics of machine health condition using an improved arima-based prediction method," in *Industrial Electronics and Applications, 2007. ICIEA 2007. 2nd IEEE Conference on*, pp. 1062–1067, 2007.

[20] W. Caesarendra, A. Widodo, P. H. Thom, B.-S. Yang, and J. Setiawan, "Combined probability approach and indirect data-driven method for bearing degradation prognostics," *Reliability, IEEE Transactions on*, vol. 60, pp. 14–20, March 2011.

[21] G. Niu and B.-S. Yang, "Intelligent condition monitoring and prognostics system based on data-fusion strategy," *Expert Systems with Applications*, vol. 37, no. 12, pp. 8831 – 8840, 2010.

[22] F. D. Maio, K. L. Tsui, and E. Zio, "Combining relevance vector machines and exponential regression for bearing residual life estimation," *Mechanical Systems and Signal Processing*, vol. 31, no. 0, pp. 405 – 427, 2012.

[23] J. Liu, W. Wang, F. Ma, Y. Yang, and C. Yang, "A data-model-fusion prognostic framework for dynamic system state forecasting," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 4, pp. 814 – 823, 2012. Special Section: Dependable System Modelling and Analysis.

[24] F. Zhao, Z. Tian, and Y. Zeng, "Uncertainty quantification in gear remaining useful life prediction through an integrated prognostics method," *Reliability, IEEE Transactions on*, vol. 62, pp. 146–159, March 2013.

[25] T. Wang, J. Yu, D. Siegel, and J. Lee, "A similarity-based prognostics approach for remaining useful life estimation of engineered systems," in *International Conference on Prognostics and Health Management, PHM 2008*, pp. 1–6, 2008.

[26] E. Ramasso, M. Rombaut, and N. Zerhouni, "Joint prediction of continuous and discrete states in time-series based on belief functions," *Cybernetics, IEEE Transactions on*, vol. 43, no. 1, pp. 37–50, 2013.

[27] L. Ye and E. Keogh, "Time series shapelets: a new primitive for data mining," in *Proc. of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pp. 947 – 956, 2009.

[28] Z. Xing, J. Pei, P. S. Yu, and K. Wang, "Extracting interpretable features for early classification on time series," in *Proc. of the Eleventh SIAM International Conference on Data Mining, SDM 2011, April 28-30, 2011, Mesa, Arizona, USA*, pp. 247–258, 2011.

[29] F. Xue, P. Bonissone, A. Varma, W. Yan, N. Eklund, and K. Goebel, "An instance-based method for remaining useful life estimation for aircraft engines," *Journal of Failure Analysis and Prevention*, vol. 8, no. 2, pp. 199–206, 2008.

[30] E. Zio and F. D. Maio, "A data-driven fuzzy approach for predicting the remaining useful life in dynamic failure scenarios of a nuclear system," *Reliability Engineering & System Safety*, vol. 95, no. 1, pp. 49 – 57, 2010.

[31] R. Khelif, S. Malinowski, B. Chebel-Morello, and N. Zerhouni, "Rul prediction based on a new similarity-instance based approach," in *Proc. of IEEE International Symposium on Industrial Electronics*, 2014.

[32] A. Saxena and K. Goebel, "C-MAPSS Data Set," in *NASA Ames Prognostics Data Repository*, 2008.

[33] T. Wang, "Trajectory similarity based prediction for remaining useful life estimation," *Thesis dissertation from University of Cincinnati*, 2010.

[34] A. Saxena, J. Celaya, B. Saha, and K. Goebel, "Metrics for offline evaluation of prognostic performance," *International Journal of Prognostics and Health Management*, April 2010.

List of Figures

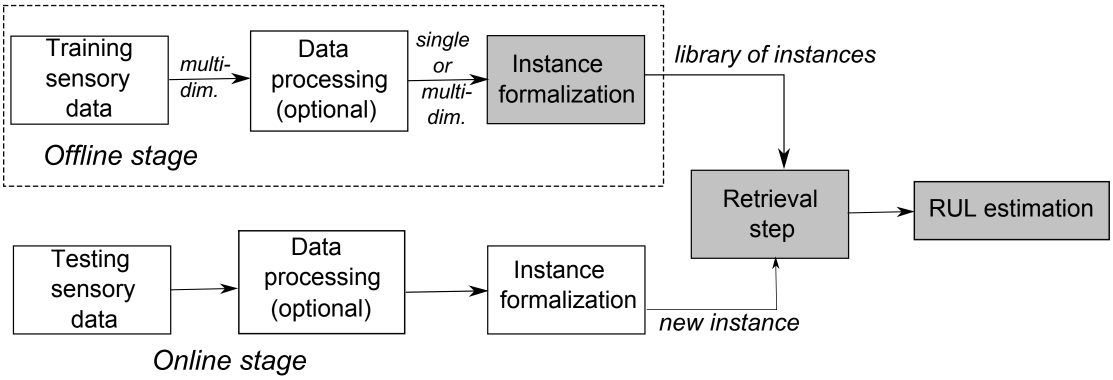

Figure 1: General framework of similarity-based RUL estimation schemes.

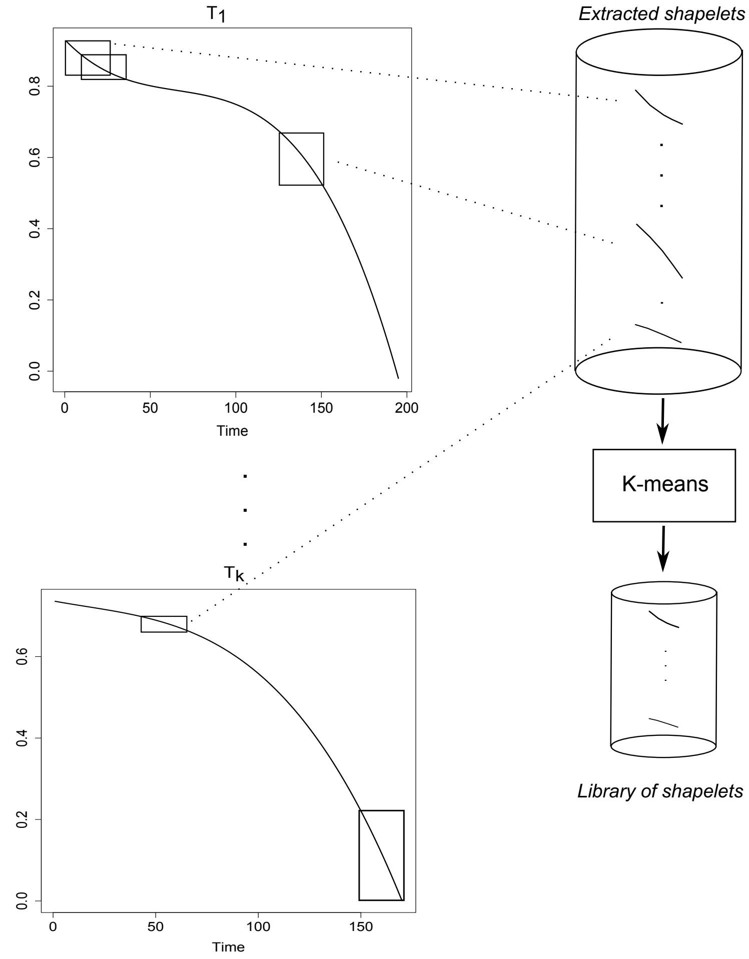

Figure 2: Shapelet extraction process : all subsequences of length l_i are extracted and stored. The number of such subsequences is then reduced by the K-means algorithm. This process is repeated for different lengths l_i if needed.

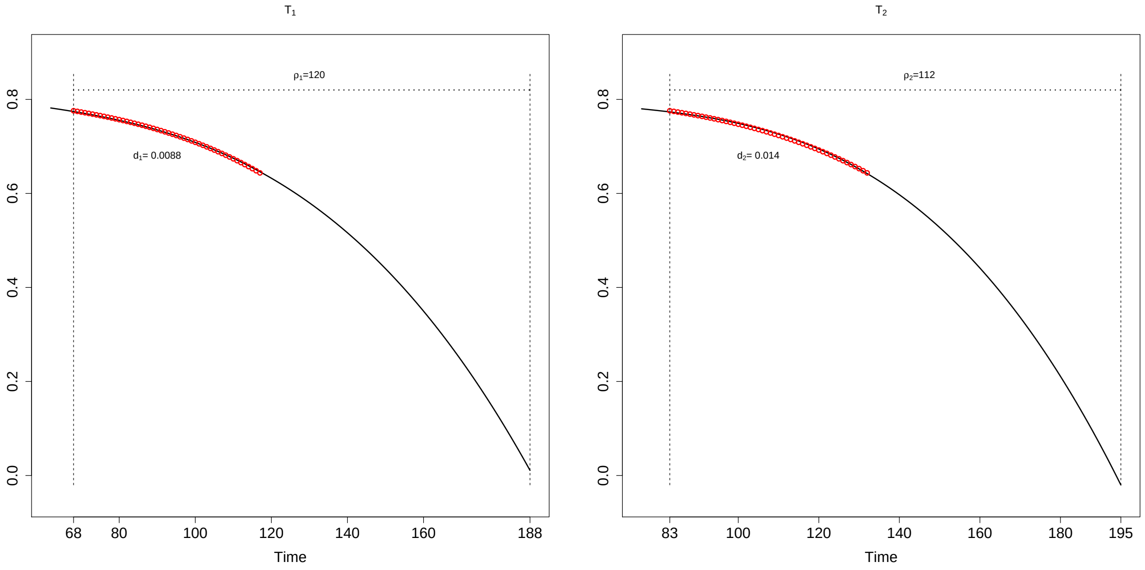

Figure 3: Shapelet selection process.

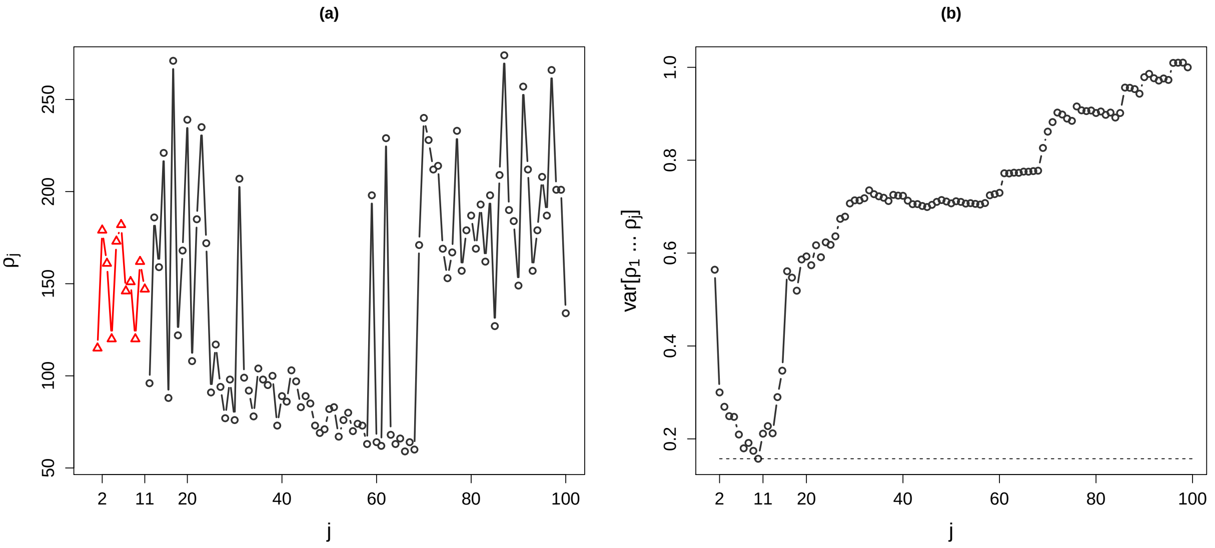

Figure 4: (a) Best-match RULs obtained between a rul-shapelet and a set of 100 training time series, ordered according to the best-match distances (Equation (2)) - (b) Partial variances of the best-match RULs of (a). The index of the minimum partial variance is equal to 11 here.

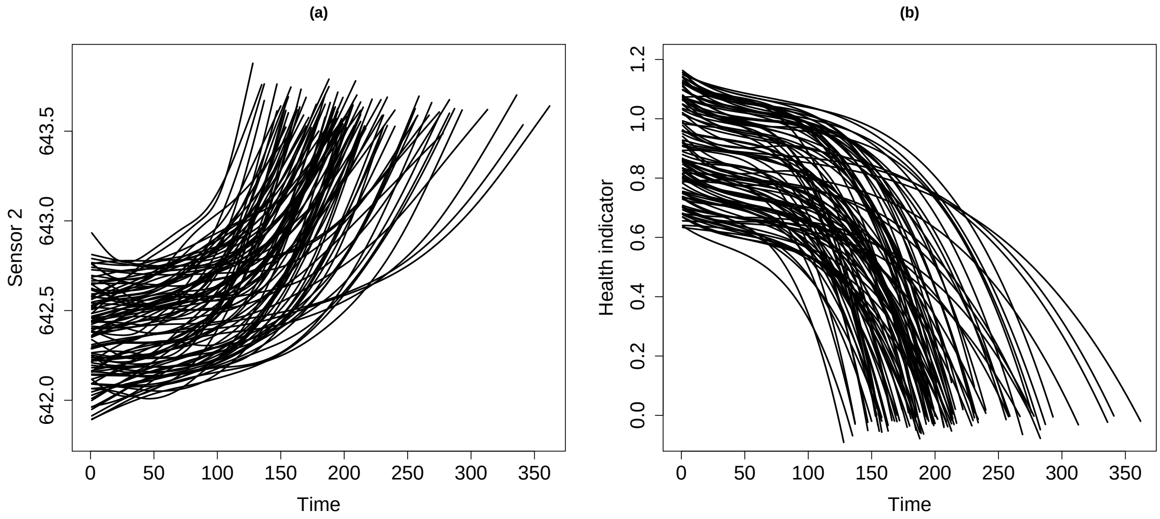

Figure 5: (a) Evolution of the measures of sensor 2 and (b) health indicators obtained by linear regression for the 100 training time series of dataset #1. These curves are obtained after smoothing with a third-order polynomial curve.

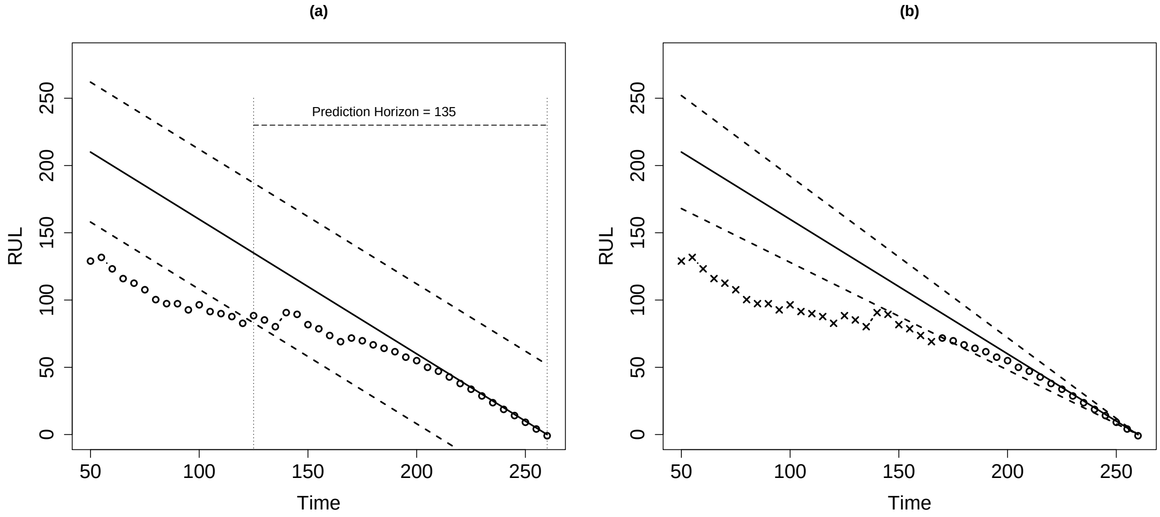

Figure 6: (a) Prediction horizon and (b) Rate of acceptable predictions.

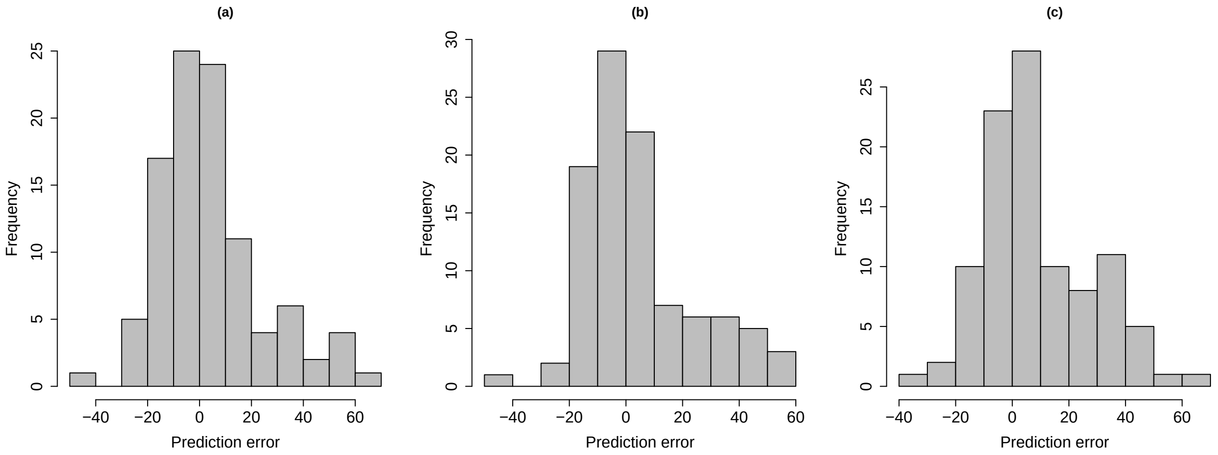

Figure 7: Histograms of the prediction errors (actual RUL − predicted RUL) obtained by applying (a) an estimation approach based on [25], the proposed approach on (b) the 7-dimensional time series and on (c) the health indicator obtained after linear regression on the original time series.

List of Tables

Table 1: Experimental settings for turbofan data sets.

Experiment number	#1	#2	#3	#4
Number of fault modes	1	1	2	2
Number of operating conditions	1	6	1	6
Number of training units	100	260	100	249
Number of testing units	100	259	100	248

Table 2: Prediction score of the proposed RUL estimation technique with different values of the parameters l_i and τ.

# centroids τ	50	100	150	200	250	300
0.15	15.35	17.19	19.39	16.97	17.82	12.34
0.25	12.36	10.30	7.12	7.38	7.36	7.64
0.35	13.30	10.94	7.19	6.88	6.87	6.75
0.45	17.49	10.34	7.00	6.80	6.71	**6.52**
0.55	17.49	10.90	7.19	6.91	6.79	6.57

Table 3: Performance evaluation for the dataset #1 in terms of the average prediction score of the proposed approach and comparison with [25].

Method	Average Prediction score of Eqn (11)
Proposed approach (7 sensors)	8.07
Proposed approach (health indicator)	**6.52**
Estimation based on Wang et al. [25]	7.91

Table 4: Performance evaluation in terms of the metrics PH, RAP and RA of the proposed approach and comparison with [33] for dataset #4.

Method	Prediction horizon	Rate of acceptable prediction	Relative accurac
Proposed approach (health indicator)	176	0.3750	0.6819
Results from Wang et al. [33]	177	0.40	0.7124

Table 5: Performance evaluation in terms of the average prediction score of the proposed approach and comparison with [25] for dataset #4.

Method	Average Prediction score of Eqn. (11)
Proposed approach (health indicator)	**37.7097**
Estimation based on Wang et al. [25]	69.2928