Title

Data-Driven Remaining Useful Life Estimation Using Gaussian Mixture Models

Permalink

https://escholarship.org/uc/item/8mj6480z

Authors

Liu, Yixuan
Hu, Zhen
Todd, Michael
et al.

Publication Date

2021-01-11

DOI

10.2514/6.2021-1487

Peer reviewed

Data-Driven Remaining Useful Life Estimation Using Gaussian Mixture Models

Yixuan Liu[1], Zhen Hu[2]
University of Michigan-Dearborn, Dearborn, MI, USA 48124

Michael Todd[3]
University of California-San Diego, La Jolla, CA, USA 92093

Chao Hu[4]
Iowa State University, Ames, IA, USA, 50011

Data-driven remaining useful life (RUL) estimation plays a vital role in system prognostics and health management (PHM). This paper presents an approach for RUL estimation based on Gaussian mixture model (GMM). A health index model is first employed to map the high-dimensional sensor signals into health index that describes the health degradation of an engineering system. Based on the mapping of health index, a Gaussian mixture model is then constructed in the time domain in the off-line phase to learn the complicated statistical relationship between the health index and RUL. The learned GMM is then used in a Bayesian scheme to perform RUL prediction in the on-line phase. The proposed approach will not only provide a point estimation of the RUL, but also a confidence interval of the RUL prediction. Using a miter gate example and the data set in 2008 PHM Data Challenge Competition as our case studies, the results show the efficacy of the proposed method.

Nomenclature

$\mathbf{C}$	=	Life cycle
e	=	Length of training set
h	=	Health index
k	=	Number of estimated parameters in the model
$\hat{L}$	=	The maximum value of the likelihood function of the model
l	=	Number of windows in selected vector of testing health index
m	=	Number of data steps in the training set
o	=	Length of testing set
$\mathbf{Q}$	=	Number of Gaussian components
$\mathbf{S}$	=	Sensor reading
s	=	Step size
$\mathbf{T}$	=	Transfer matrix
t	=	Remaining useful life
u	=	Standardized health index
w	=	Window size
$\mathbf{Z}$	=	Input data for Gaussian mixture model
η	=	Smoothing parameter
θ	=	Standardized remaining useful life
λ	=	Weight of Gaussian Mixture Model

[1] Ph.D. student, Dept. of Industrial and Manufacturing Systems Eng., Dearborn, MI, 48128.
[2] Assistant Professor, Dept. of Industrial and Manufacturing Systems Eng., Dearborn, MI, 48128., AIAA Member.
[3] Professor, Department of Structural Engineering, University of California-San Diego, La Jolla, CA, 92093.
[4] Assistant Professor, Dept. of Mechanical Engineering, Iowa State University, Ames, IA, 50011, AIAA Member.

μ	=	Mean value for Gaussian mixture model
$\mathbf{\Sigma}$	=	Covariance matrix of Gaussian mixture model
Ω	=	Training data set

I. Introduction

Remaining useful life (RUL) is drawing great interest in various research areas [1], such as electronics [2], mechanics [3], aircraft industry [4], medicine [5], and even weather forecast [6]. According to its definition, the remaining useful life of a system is defined as the distance between the current time and the end life of a system (useful life) [7]. RUL estimation plays an important role in condition-based maintenance [8, 9]. It provides information that is vital for maintenance planning and reduce the overall lifecycle cost [10, 11].

Because of the importance of RUL estimation, numerous approaches have been proposed in recent years. For example, Le Son et al. has combined Wiener process with Principal Component Analysis and proposed a probabilistic approach for RUL prediction [12]. Cui et al. [13] proposed a new switching unscented Kalman filter algorithm to predict the RUL of bearing using its Condition Monitoring (CM) data. Xi et al. [14] established a copula-based sampling method with offline training process and online prediction process for RUL estimation. Si et al. [15] introduced an approach with the combination of recursive filter, expectation maximization (EM) algorithm, and Wiener process to update the degradation model for RUL. Heimes et al. [16] used a recurrent neural network with extended Kalman filter for RUL estimation with given system monitoring data. Li et al. [17] proposed an approach using deep convolution neural networks (DCNN) for RUL estimation. Similarly, Zhang et al. [18] proposed a multi-objective deep belief networks ensemble (MODBNE) method to keep a balance between accuracy and diversity during RUL prediction. Babu et al. [19] suggested another deep Convolutional Neural Network (CNN) approach RUL estimation.

Even though many approaches have been proposed in recent year for RUL estimation, there is no universal approach that is applicable to RUL estimation. Each method has its own advantages and disadvantages. For instance, some approaches are too complicated to be implemented by engineers in practical applications; some approaches are computationally very expensive for online prediction which cannot be deployed in practical systems without high computational power; and some approaches can only provide a point estimation of the RUL. This paper aims to overcome limitations of some of the existing approaches by developing a Gaussian mixture model (GMM)-based RUL method. By taking advantage of the capability of GMM in learning the complex joint probability density function using data, a GMM-based approach is proposed in this paper, which includes an off-line phase and an on-line phase. In the off-line phase, a GMM is learned based on health index and RUL data. The GMM is then used under a Bayesian updating scheme to improve the prediction accuracy of RUL. Two examples, including a practical application and the 2008 PHM Data Challenge problem are employed to verify the effectiveness of the proposed approach.

The remainder of this paper is organized as follows: Section 2 introduces the background of RUL Prediction and GMM. Section 3 describes the proposed approach for RUL estimation. Section 4 uses an engineering application and the PHM08 datasets to demonstrate the effectiveness of the proposed method. Section 5 concludes this paper and discuss future work.

II. Background

In this section, we first briefly review the approach used to construct health index. After that, we provide an overview of GMM which will be used to develop the proposed method.

A. Construction of the degradation health index

To monitor and diagnose the condition of a system, many sensors are used to collect the data of a system. Since the data is multi-dimensional and different sensor data is showing different patterns, in order to better assess the degradation of the system and estimate the remaining useful life, the dimension of the data need to be reduced (i.e. into one dimension) [22]. There are different ways to fulfill it. For instance, Mosallam et al. [23] used principal component analysis approach to transform the data form a high-dimensional space into a single-dimensional space. The transformed single-dimensional signal is called health index [22] or health indicator (HI) [15]. Similarly, Benkedjouh et al. [24] performed the isometric mapping technique (ISOMAP) as the feature reduction approach to generate a one-dimensional health index. Both methods have one short come that they are losing too much information in the feature reduction process. To maintain the integrity of the data, Yan et al. [25] used logistic regression to convert the multi-dimensional sensor readings into one-dimensional health index. However, logistic regression is less sensitive near the early and end of a system life and the original pattern of the sensor is deformed too much [15]. As an improved version, Wang et. al [15] used linear regression instead of logistic regression to generate health index. In both methods, health index is constrained between 0 and 1, which respectively represents the beginning and the end of system life.

The method for health index construction we use is similar to the one as discussed by Wang et. [15]. Figure. 1 shows an example of fusing multi-dimensional sensor readings into a one-dimensional health index.

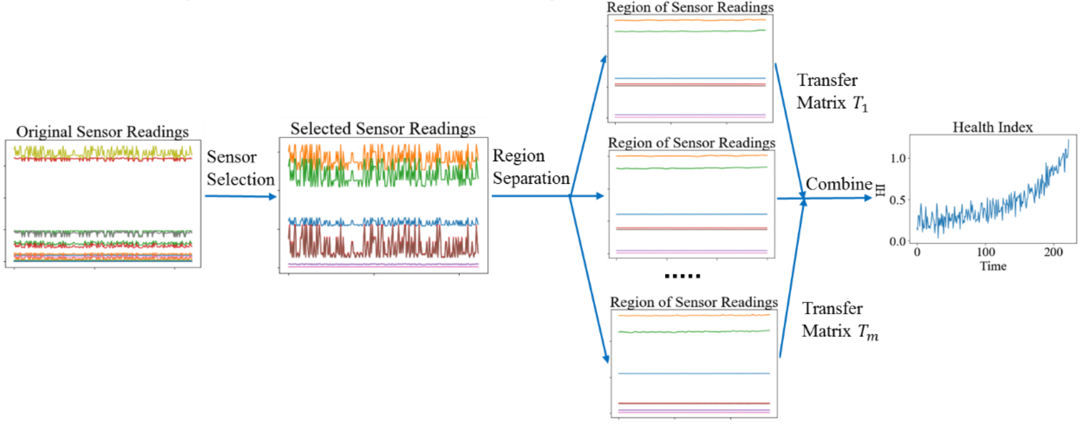

Figure. 1 An example of constructing health index from sensor readings

After the multi-dimensional sensor data is selected, it will be used to train the linear regression model. The sensor data are related to the HI as follows [20]

$$\mathbf{h}_{tr} = \mathbf{S}_{tr} \times \mathbf{T} \text{ or } \mathbf{h}_{te} = \mathbf{S}_{te} \times \mathbf{T} \tag{1}$$

where **T** is a transfer matrix, $\mathbf{S}_{tr}$ and $\mathbf{S}_{te}$ are respectively the training and testing sensor readings, and $\mathbf{h}_{tr}$ and $\mathbf{h}_{te}$ are respectively the health indices of the training and testing data.

In order to get the transfer matrix **T**, a particular training data set Ω is selected as follows [20]

$$\Omega = \mathbf{S}_1 \cup \mathbf{S}_0 = \{\mathbf{S}_{off} \mid C > C_{end}\} \cup \{\mathbf{S}_{off} \mid C < C_{start}\}, \tag{2}$$

where C is the time cycle, $C_{\max}$ and $C_{\min}$ respectively represents the early life and the end life of the system.

After having $\mathbf{S}_0$ and $\mathbf{S}_1$, the transfer matrix $\mathbf{T}$ is calculated by

$$\mathbf{T} = (\mathbf{S}^T\mathbf{S})^{-1}\mathbf{S}^T\mathbf{V}, \tag{3}$$

where $\mathbf{S} = [\mathbf{S}_1; \mathbf{S}_0]$, $\mathbf{V} = [\mathbf{V}_1, \mathbf{V}_0]^T$, $\mathbf{V}_1$ is a unity vector that has the same column length as $\mathbf{S}_1$, $\mathbf{V}_0$ is a zero vector that has the same column length as $\mathbf{S}_0$ [14].

B. Gaussian mixture model

Gaussian Mixture Model (GMM) is an method using a weighted sum of a number of Gaussian components to represent one parametric probability density function (PDF) [21]. This method has been commonly applied to analyze features from a complex system, such as speaker recognition [22] or background subtraction [23].

For a random variable *X*, a GMM can be used to approximate its PDF as follow

$$f_{X_i}(x) = \sum_{i=i}^{Q} \lambda_i \phi(x, \mu_i, \sigma_i^{\,2}), \tag{4}$$

where Q is the number of Gaussian components, $\phi(\cdot)$ is the PDF of a Gaussian random variable, λ_i, μ_i and σ_i are respectively the weight, mean, and standard deviation of the *i*-th Gaussian component.

If $\mathbf{X} \in \mathbb{R}^a$, $\mathbf{Y} \in \mathbb{R}^b$, then for $\mathbf{Z} = [\mathbf{X}, \mathbf{Y}] \in \mathbb{R}^{a+b}$, the joint PDF $f_{\mathbf{X},\mathbf{Y}}(\mathbf{X}, \mathbf{Y})$ can be estimated using a multi-variable GMM as

$$f_{\mathbf{Z}}(\mathbf{Z}) = \sum_{i=i}^{Q} \lambda_i \phi(\mathbf{z}, \boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i), \tag{5}$$

where $\boldsymbol{\mu}_i = [\boldsymbol{\mu}_{i,\mathbf{X}}, \boldsymbol{\mu}_{i,\mathbf{Y}}]$ and $\boldsymbol{\Sigma}_i = \begin{bmatrix} \boldsymbol{\Sigma}_{i,\mathbf{XX}} & \boldsymbol{\Sigma}_{i,\mathbf{XY}} \\ \boldsymbol{\Sigma}_{i,\mathbf{YX}} & \boldsymbol{\Sigma}_{i,\mathbf{YY}} \end{bmatrix}$ are the mean and covariance matrices of the *i*-th Gaussian component, in which $\boldsymbol{\mu}_{i,\mathbf{X}}$ and $\boldsymbol{\mu}_{i,\mathbf{Y}}$ are mean vectors and $\boldsymbol{\Sigma}_{i,\mathbf{XX}}$, $\boldsymbol{\Sigma}_{i,\mathbf{XY}}$, $\boldsymbol{\Sigma}_{i,\mathbf{YX}}$, and $\boldsymbol{\Sigma}_{i,\mathbf{YY}}$ are covariance matrices.

III. Proposed Approach

In this section, we present our proposed GMM-based RUL estimation approach.

A. Overview

Our research aims to improve the RUL estimation in the following ways. First, we want to provide a distribution of the RUL estimation instead of a single value. Second, we want to consider the correlation between the nearby data. Inspired by Ref. [24], we propose a GMM-based approach for RUL estimation. Fig. 2 shows the flowchart of the proposed method. The basic idea of this approach is by using GMM in conjunction with Bayesian inference for RUL estimation and updating. As shown in this figure, this approach also has an offline training phase and an online prediction phase. In the offline training phase, the data will go through a series of pre-processing steps and then train a GMM. In the online prediction phase, distribution of the RUL will be calculated using the testing health index with the trained GMM and Bayesian inference. In what follows, we explain the proposed method in details.

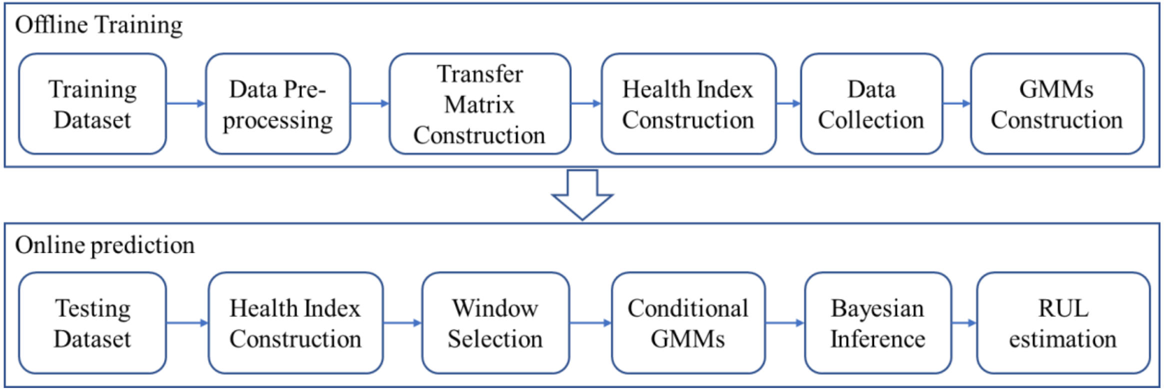

Figure.2 Flowchart of GMM based approach

B. Data processing

In the single GMM–based approach, the prior sample remains the same all the time, so the uncertainty of the prediction is not reduced over time during the prediction phase. In order to overcome this problem, we propose another approach by coupling two GMMs under a Bayesian scheme. More specially, one GMM model is used to estimate the RUL posterior distribution at each time step and the other GMM is used to update the prior distribution of RUL for Bayesian inference.

Fig. 3 shows the flowchart of data processing. Using the procedure in Section 2.1, we will be able to transfer sensor readings into health index **h** and its corresponding RUL **t**. Before collecting data to build the Gaussian mixture model, we transform the original data **h** and **t** into standard normal domain. This step is called data standardization. The reason to implement this step is to make sure that the joint PDF of the variables can be approximated as mixture of Gaussian components [24], which can increase the accuracy of GMM prediction.

For the data standardization, we have $[\mathbf{t},\mathbf{h}]^T = \begin{bmatrix} t & t_2 & \cdots & t_e \\ h_1 & h_2 & \cdots & h_e \end{bmatrix}$ for each training set, where e is the number of data points in the training set, knowing that e is different among different training set.

The data **h** and **t** can then be transformed into standard normal domain as follows

$$\theta_i = \Phi^{-1}(F_T(t_i)), \forall i = 1, \cdots, e, \tag{6}$$

$$u_i = \Phi^{-1}(F_H(h_i)), \forall i = 1, \cdots, e, \tag{7}$$

where is Φ^{-1} is the inverse cumulative distribution function (CDF) of a standard normal variable, $F_H(\cdot)$ is the CDF of the training health index and $F_T(\cdot)$ is the CDF of the training RUL. $F_H(\cdot)$ and $F_T(\cdot)$ are calculated using the kernel density estimation with Gaussian kernel.

$$F_H(h) = \frac{1}{e\eta_H}\sum_{i=1}^{e} K(\frac{h-h_i}{\eta_H}) \tag{8}$$

$$F_T(t) = \frac{1}{e\eta_T}\sum_{i=1}^{e} K(\frac{t-t_i}{\eta_T}) \tag{9}$$

where $K(\cdot)$ is the kernel function and $K(\cdot)=\Phi(\cdot)$ for Gaussian kernel, η_H and η_T are the smoothing parameters called bandwidth.

After data standardization, we have $[\boldsymbol{\theta},\mathbf{u}]^T=\begin{bmatrix}\theta_1 & \theta_2 & \cdots & \theta_e\\ u_1 & u_2 & \cdots & u_e\end{bmatrix}$ for each training set. Then we can then collect the data to build the GMM using a moving window with a window size of *2w* and a step size of *s*, where 2*w* means the number of health index collected in one vector, *w* will also be used as the window size in the online prediction phase, *s* means the distance between two nearby successive windows. Fig. 3 illustrates a general procedure of the data collection process.

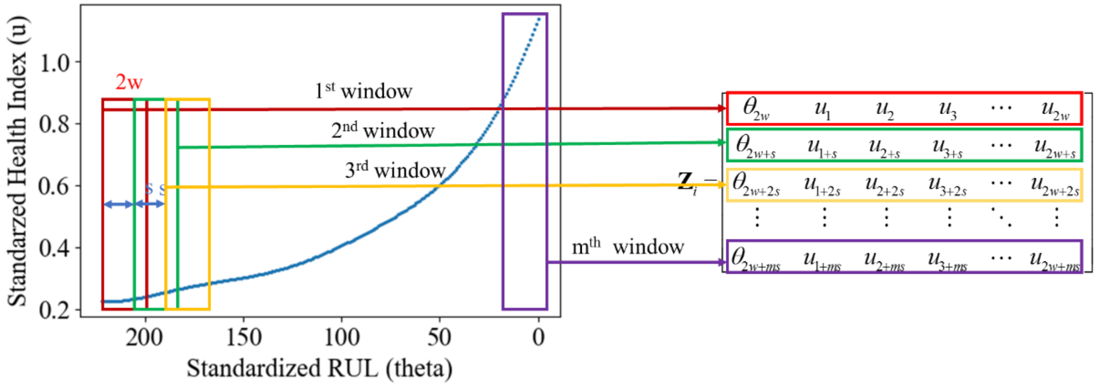
Standarized Health Index (u)
1.0
0.8
0.6
0.4
0.2
200
150
100
50
0
Standardized RUL (theta)
2w
s
s
1st window
2nd window
3rd window
mth window
$\mathbf{Z}_i=$
θ_{2w} u_1 u_2 u_3 $\cdots$ u_{2w}
θ_{2w+s} u_{1+s} u_{2+s} u_{3+s} $\cdots$ u_{2w+s}
θ_{2w+2s} u_{1+2s} u_{2+2s} u_{3+2s} $\cdots$ u_{2w+2s}
θ_{2w+ms} u_{1+ms} u_{2+ms} u_{3+ms} $\cdots$ u_{2w+ms}

Figure.3 Data collection for GMM

Through the data processing using the sliding window, $[\boldsymbol{\theta},\mathbf{u}]^T$ of the *i*-th training signal becomes to be a matrix as follow

$$\mathbf{Z}_i=[\mathbf{z}_1,\mathbf{z}_2,\cdots,\mathbf{z}_m]^T=\begin{bmatrix}\theta_{2w} & u_1 & u_2 & u_3 & \cdots & u_{2w}\\ \theta_{2w+s} & u_{1+s} & u_{2+s} & u_{3+s} & \cdots & u_{2w+s}\\ \theta_{2w+2s} & u_{1+2s} & u_{2+2s} & u_{3+2s} & \cdots & u_{2w+2s}\\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots\\ \theta_{2w+ms} & u_{1+ms} & u_{2+ms} & u_{3+ms} & \cdots & u_{2w+ms}\end{bmatrix}, \tag{10}$$

where *m* is the number of steps in this training set, knowing that *m* is different among different training sets and $2w+m\le e$ for each training set.

If we have *N* groups of $[\boldsymbol{\theta},\mathbf{u}]^T$ pair, the sample data **Z** will become

$$\mathbf{Z}=\begin{bmatrix}\mathbf{Z}_1\\ \mathbf{Z}_2\\ \vdots\\ \mathbf{Z}_N\end{bmatrix}, \tag{11}$$

where $\mathbf{Z}_i$ is collected using Eq. (10).

C. Training of Gaussian mixture model

For the offline training phase, the data pre-processing part is the same as the single GMM-based approach. A GMM is constructed based on the health index data in the time domain. For the online prediction phase, Fig. 4 gives an overview of the overall process. For the first step, we still choose the first window of HI $\mathbf{H}_w^0=[h_1,h_2,\cdots,h_w]\in\mathbf{H}_{test}$, by applying Eq. (9) to (12), we will be able to calculate the PDF function of the RUL for the first step. For the next step, Bayesian Inference will be applied to update the distribution of RUL as follows

Once all the training data are collected, a Gaussian Mixture model can be constructed as below using the expected maximization methods [25]. A trained GMM will have the PDF function written as

$$f_{\mathbf{Z}}(\mathbf{Z})=\sum_{i=i}^{Q}\lambda_i\phi(\mathbf{z},\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i), \tag{12}$$

where $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ are given by

$$\boldsymbol{\mu}_i = [\mu_1, \mu_2, \mu_3, \cdots, \mu_{2w+1}], \tag{13}$$

$$\boldsymbol{\Sigma}_i = \begin{bmatrix} \sigma_1^2 & \omega_{1,2} & \omega_{1,3} & \cdots & \omega_{1,2w+1} \\ \omega_{2,1} & \sigma_2^2 & \omega_{2,3} & \cdots & \omega_{2,2w+1} \\ \omega_{3,1} & \omega_{3,2} & \sigma_3^2 & \cdots & \omega_{3,2w+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \omega_{2w+1,1} & \omega_{2w+1,2} & \omega_{2w+1,3} & \cdots & \sigma_{2w+1}^2 \end{bmatrix} \tag{14}$$

where $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ are the GMM parameters, μ_i represents the mean value of the i^{th} element, σ_i^2 represents the variance of the i^{th} element and $\omega_{i,j}$ represents the covariance between the i^{th} and the j^{th} elements.

According to the format of **Z** in Eq. (10), the relationship between the variables of HI and RUL and the parameters are shown in Fig. 4. μ_1 and stands for the mean value of RUL in GMM, $[\mu_2, \cdots, \mu_{2w+1}]$ stands for the mean value of two nearby windows of HIs. Using these relationships, $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$ can be rewritten as follows

$$\boldsymbol{\mu}_i = [\mu_{i,\theta}, \cdots, \mu_{i,u}] = [\mu_{i,\theta}, \boldsymbol{\mu}_{i,1}, \boldsymbol{\mu}_{i,2}] = [\boldsymbol{\mu}_{i,0}, \boldsymbol{\mu}_{i,2}], \tag{15}$$

where $\mu_{i,\theta} = \mu_1$, $\mu_{i,u} = \mu_{2w+1}$, $\boldsymbol{\mu}_{i,1} = [\mu_2, \cdots, \mu_{w+1}]$, $\boldsymbol{\mu}_{i,2} = [\mu_{w+2}, \cdots, \mu_{2w+1}]$, $\boldsymbol{\mu}_{i,0} = [\mu_1, \boldsymbol{\mu}_{i,1}]$

$$\boldsymbol{\Sigma}_i = \begin{bmatrix} \sigma_{i,\theta}^2 & \cdots & \omega_{i,\theta u} \\ \vdots & \ddots & \vdots \\ \omega_{i,u\theta} & \cdots & \sigma_{i,u}^2 \end{bmatrix} = \begin{bmatrix} \sigma_{i,\theta}^2 & \boldsymbol{\Sigma}_{i,\theta 1} & \boldsymbol{\Sigma}_{i,\theta 2} \\ \boldsymbol{\Sigma}_{i,1\theta} & \boldsymbol{\Sigma}_{i,11} & \boldsymbol{\Sigma}_{i,12} \\ \boldsymbol{\Sigma}_{i,2\theta} & \boldsymbol{\Sigma}_{i,21} & \boldsymbol{\Sigma}_{i,22} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\Sigma}_{i,00} & \boldsymbol{\Sigma}_{i,02} \\ \boldsymbol{\Sigma}_{i,20} & \boldsymbol{\Sigma}_{i,22} \end{bmatrix}, \tag{16}$$

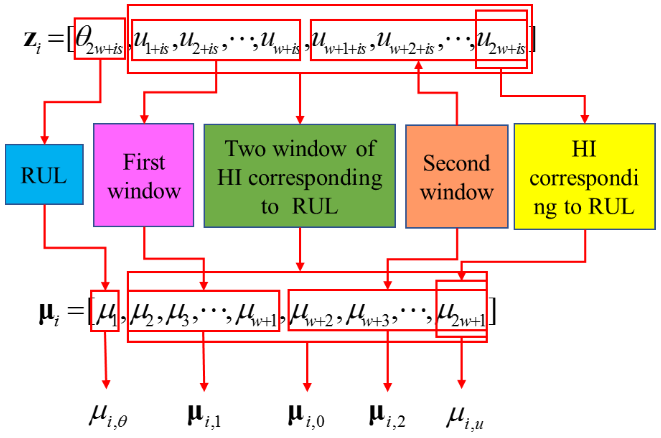
$\mathbf{z}_i = [\theta_{2w+is}, u_{1+is}, u_{2+is}, \cdots, u_{w+is}, u_{w+1+is}, u_{w+2+is}, \cdots, u_{2w+is}]$
RUL
First window
Two window of HI corresponding to RUL
Second window
HI correspondi ng to RUL
$\boldsymbol{\mu}_i = [\mu_1, \mu_2, \mu_3, \cdots, \mu_{w+1}, \mu_{w+2}, \mu_{w+3}, \cdots, \mu_{2w+1}]$
$\mu_{i,\theta}$
$\boldsymbol{\mu}_{i,1}$
$\boldsymbol{\mu}_{i,0}$
$\boldsymbol{\mu}_{i,2}$
$\mu_{i,u}$

Figure.4 Relationship between variables and parameters

where $\sigma_{i,\theta}^2 = \sigma_1^2$, $\sigma_{i,u}^2 = \sigma_{2w+1}^2$, $\omega_{i,\theta u} = \omega_{i,u\theta} = \omega_{1,2w+1}$, $\boldsymbol{\Sigma}_{i,11} = \begin{bmatrix} \sigma_2^2 & \cdots & \omega_{2,w+1} \\ \vdots & \ddots & \vdots \\ \omega_{w+1,2} & \cdots & \sigma_{w+1}^2 \end{bmatrix}$, $\boldsymbol{\Sigma}_{i,22} = \begin{bmatrix} \sigma_{w+2}^2 & \cdots & \omega_{w+2,2w+1} \\ \vdots & \ddots & \vdots \\ \omega_{2w+1,w+2} & \cdots & \sigma_{2w+1}^2 \end{bmatrix}$, $\boldsymbol{\Sigma}_{i,00} = \begin{bmatrix} \sigma_{i,\theta}^2 & \boldsymbol{\Sigma}_{i,\theta 1} \\ \boldsymbol{\Sigma}_{i,1\theta} & \boldsymbol{\Sigma}_{i,11} \end{bmatrix}$.

The parameter pair $[\mu_{i,\theta}, \mu_{i,u}]$ and $\begin{bmatrix} \sigma_{i,\theta}^2 & \omega_{i,\theta u} \\ \omega_{i,u\theta} & \sigma_{i,u}^2 \end{bmatrix}$ represent the relationship between RUL and a single HI, and it will called the *first* part of the GMM parameters in the proposed approach. $[\mu_{i,\theta}, \boldsymbol{\mu}_{i,2}]$ and $\begin{bmatrix} \sigma_{i,\theta}^2 & \boldsymbol{\Sigma}_{i,\theta 2} \\ \boldsymbol{\Sigma}_{i,2\theta} & \boldsymbol{\Sigma}_{i,22} \end{bmatrix}$ represent the relationship between RUL and the corresponding window of HIs, and it is called the *second* part of the GMM parameters which will be used to estimate RUL in Sec. 3.3. $[\boldsymbol{\mu}_{i,0},\boldsymbol{\mu}_{i,2}]$ and $\begin{bmatrix}\boldsymbol{\Sigma}_{i,00} & \boldsymbol{\Sigma}_{i,02}\\ \boldsymbol{\Sigma}_{i,20} & \boldsymbol{\Sigma}_{i,22}\end{bmatrix}$ represent the relationship between RUL and the corresponding two nearby windows of HIs, and it is called the whole GMM parameters.

Fig. 5 shows how different parts of parameters is located in $\boldsymbol{\mu}_i$ and $\boldsymbol{\Sigma}_i$.

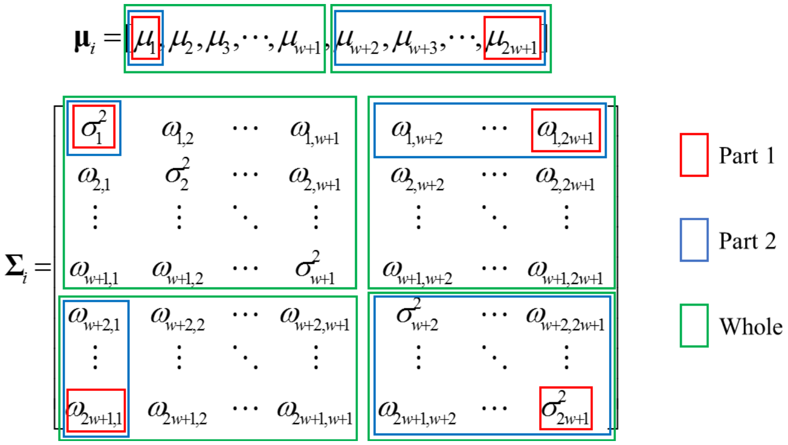

Figure.5 Locations of different parts of parameters

When training the GMM, the number of Gaussian components *Q* will affect the value of parameters in the multivariate Gaussian distribution, and lead to different results in the prediction phase. In order to find the best GMM with a proper *Q*, we use can Akaike information criterion (AIC) [26] or Bayesian information criterion (BIC)[27].

The AIC value for a statistical model is given as [28]

$$AIC = 2k - 2\ln(\hat{L}) \tag{17}$$

where *k* is the number of estimated parameters in the model, $\hat{L}$ is the maximum value of the likelihood function of the model.

The BIC value for a model is defined as [29]

$$BIC = k\ln(n) - 2\ln(\hat{L}) \tag{18}$$

where *n* is the number of data points or the sample size.

In order to find the best GMM, we construct different GMM with different *Q* value, them calculate the AIC or BIC value of each model using Eq. (17) and (18). The model that has the lowest value is the best model. After the best GMM is obtained from the offline training phase, we can go to online prediction phase to use the obtained GMM for RUL prediction.

D. RUL estimation using GMM and Bayesian updating

For a given set of testing sensor readings, we first transform the sensor readings into health index using Eq. (1). After that, we transform the testing health index into standard normal domain. For a given group of testing health index set $\mathbf{h}_{te} = [h_1, h_2, \cdots, h_o]$, it can be standardized using Eq. (7), then we have $\mathbf{u}_{te} = [u_1, u_2, \cdots, u_o]$, where *o* is the number of health index in the test set. After the transformation, the RUL (i.e. t_T) for this particular set of health index $\mathbf{u}_{se} = [u_1, u_2, \cdots, u_{ind}] \in \mathbf{u}_{te}$ is calculated using Bayesian method as below

$$f(t_T \mid \mathbf{u}_{se}) = \frac{f(\mathbf{u}_{se} \mid t_T)\, f(t_T)}{\int f(\mathbf{u}_{se} \mid t_T)\, f(t_T)dt} \propto f(\mathbf{u}_{se} \mid t_T)\, f_T(t_T), \tag{19}$$

where "$\propto$" stands for "proportional to", $f_T(t_T)$ is the prior distribution of RUL and $f(\mathbf{u}_{se} \mid t_T)$ is the likelihood function of health index with given RUL.

Since the length of testing health index is usually long and monitoring data will be continuously collected in the online phase, to address this issue, a sliding window with a window size of *w* is employed to extract subsets of the health index from the data. Based on that, we can partition the testing health index into many subsets as follows.

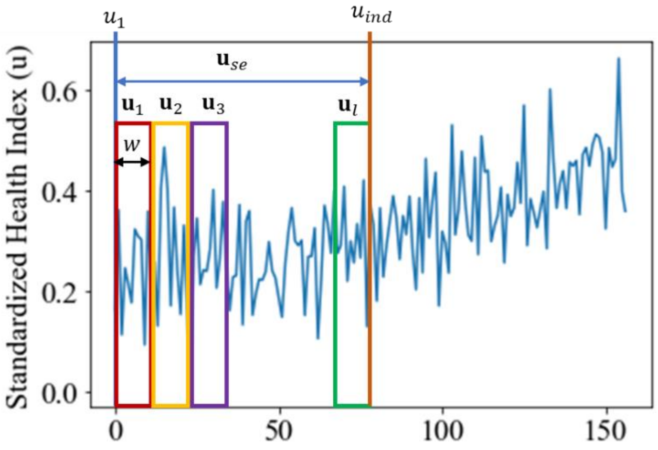

Figure.6 Relationship between $\mathbf{u}_i$ and $\mathbf{u}_{se}$

Since $\mathbf{u}_{se} = [u_1, u_2, \cdots, u_{ind}] = \mathbf{u}_1 \cup \mathbf{u}_2 \cdots \mathbf{u}_l$, where $\mathbf{u}_i$ are the windows of HI included in $\mathbf{u}_{se}$, l is the number of windows contained in $\mathbf{u}_{se}$ shown in Fig. 6, each window has the same window size *w*, which is the length of $\mathbf{u}_i$. After the partition of the health index, we have $f(\mathbf{u}_{se} \mid t_T)$ as

$$f(\mathbf{u}_{se} \mid t_T) = \prod_{k=1}^{l} f(\mathbf{u}_k \mid t_T), \tag{20}$$

If we consider the correlation between the two nearby windows, Eq. (20) becomes

$$f(\mathbf{u}_{se} \mid t_T) = f(\mathbf{u}_1 \mid t_T) \prod_{k=1}^{l-1} f(\mathbf{u}_{k+1} \mid \mathbf{u}_k, t_T), \tag{21}$$

As a result, Eq. (19) becomes

$$f(t_T \mid \mathbf{u}_{se}) \propto f(\mathbf{u}_1 \mid t_T) \prod_{k=1}^{l-1} f(\mathbf{u}_{k+1} \mid \mathbf{u}_k, t_T) f_T(t_T). \tag{22}$$

Also, the posterior distribution of standardized RUL (θ) can be written as

$$f(\theta \mid \mathbf{u}_{se}) \propto f(\mathbf{u}_1 \mid \theta) \prod_{k=1}^{l-1} f(\mathbf{u}_{k+1} \mid \mathbf{u}_k, \theta) f_\theta(\theta). \tag{23}$$

As shown in the above equations, to perform Bayesian updating of the RUL, there are three parts that need to be calculated, namely

1. The prior distribution of RUL $f_T(t_T)$
2. The likelihood function for the first window of health index $f(\mathbf{u}_1 \mid t_T)$
3. The likelihood function of the two nearby windows of health index $\prod_{k=1}^{l-1} f(\mathbf{u}_{k+1} \mid \mathbf{u}_k, t_T)$.

In this paper, we use the GMM trained in Sec. III, part C to compute the above three terms.

1. Calculation of the prior distribution $f_T(t_T)$

The prior distribution is calculated differently for different steps in order to increase prediction accuracy. For the first step, we have $\mathbf{u}_1 \in \mathbf{u}_{te}$, the prior distribution is calculated using the GMM. We use the beginning and the end of the standardized health index ($u_1, u_o \in \mathbf{u}_{te}$), and calculated the conditional GMM using the following equations

$$f_{\theta|u_1}(\theta \mid u_1) = \sum_{i=i}^{Q} \lambda_i^{(1)}(u_1) \phi(\theta, \mu_i^{(1)}(u_1), \sigma_i^{2(1)}), \tag{24}$$

$$f_{t|u_{n\_test}}(\theta \mid u_o) = \sum_{i=i}^{Q} \lambda_i^{(1)}(u_o) \phi(\theta, \mu_i^{(1)}(u_o), \sigma_i^{2(1)}), \tag{25}$$

where the elements in Eq. (23) and (24) are calculated with the parameters from the first part of the GMM using

$$\mu_i^{(1)}((u_1 \text{or } u_o)) = \mu_{i,\theta} + \omega_{i,\theta u}\sigma_{i,u}^2((u_1 \text{or } u_o) - \mu_{i,u}), \tag{26}$$

$$\sigma_i^{2(1)} = \sigma_{i,\theta}^2 - \omega_{i,\theta u}(\sigma_{i,u}^2)^{-1}\omega_{i,u\theta}, \tag{27}$$

$$\lambda_i^{(1)}(u_1 \text{ or } u_o) = \frac{\lambda_i\phi((u_1 \text{ or } u_o), \mu_{i,u}, \sigma_{i,u}^2)}{\sum_{k=1}^{Q}\lambda_k\phi((u_1 \text{ or } u_o), \mu_{k,u}, \sigma_{k,u}^2)}, \tag{28}$$

After we obtain $f_{\theta|u_1}(\theta \,|\, u_1)$ and $f_{t|u_o}(\theta \,|\, u_o)$, the prior distribution of t_T in the standard normal domain is calculated using

$$f_\theta(\theta) = g(f_{t|u_1}(\theta \,|\, u_1), f_{t|u_o}(\theta \,|\, u_o)), \tag{29}$$

where $g(\cdot)$ is a function combining the two different distributions.

We then convert $f(\theta)$ into $f_T(t_T)$ as

$$f_T(t_T) = f_\theta(\Phi^{-1}(F_T(t_T))), \tag{30}$$

where $F_T(\cdot)$ is the CDF of the marginal RUL distribution obtained from the GMM model.

For the second step, we use

$$f_T(t_T) = g(f(t_T \,|\, \mathbf{u}_1) - 2w, f(t_T \,|\, \mathbf{u}_1) - w) \tag{31}$$

where $f(t_T \,|\, \mathbf{u}_1)$ is the posterior distribution calculated from the first step, *w* is the window size.

For the rest steps, we use

$$f_T(t_T) = g(f(t_T \,|\, \mathbf{u}_{k-2}) - 2w, f(t_T \,|\, \mathbf{u}_{k-1}) - w), \tag{32}$$

where $f_{prior}(t_T \,|\, \mathbf{u}_{k-1})$ is the prior distribution form the last step, $f_{post}(t_T \,|\, \mathbf{u}_{k-2})$ is the posterior distribution calculated from the two steps before.

2. Calculation of $f(\mathbf{u}_1 \,|\, t_T)$

Similar to $f_T(t_T)$, the likelihood function $f(\mathbf{u}_1 \,|\, t_T)$ is calculated from conditional GMM as follows

$$f(\mathbf{u}_1 \,|\, t_T) = \sum_{i=i}^{Q}\lambda_i^{(2)}(\theta)\phi(\mathbf{u}, \boldsymbol{\mu}_i^{(2)}, \boldsymbol{\Sigma}_i^{(2)}) \tag{33}$$

where $\theta = \Phi^{-1}(F_T(t_T))$, the mean vector and covariance matrix in Eq. (30) are calculated with the parameters from the second part of the GMM using

$$\boldsymbol{\mu}_i^{(2)} = \boldsymbol{\mu}_{i,2} + \boldsymbol{\Sigma}_{i,\theta 2}(\sigma_{i,\theta}^2)^{-1}(\theta - \mu_{i,\theta}), \tag{34}$$

$$\boldsymbol{\Sigma}_i^{(2)} = \boldsymbol{\Sigma}_{i,22} - \boldsymbol{\Sigma}_{i,2\theta}(\sigma_{i,\theta}^2)^{-1}\boldsymbol{\Sigma}_{i,\theta 2}, \tag{35}$$

$$\lambda_i^{(2)}(\theta) = \frac{\lambda_i\phi(\theta, \mu_{i,\theta}, \sigma_{i,\theta}^2)}{\sum_{k=1}^{Q}\lambda_k\phi(\theta, \mu_{i,\theta}, \sigma_{i,\theta}^2)}, \tag{36}$$

With the prior distribution and likelihood function, we will be able to calculate the posterior distribution using Bayesian inference.

3. Calculation of $f(\mathbf{u}_{k+1} \,|\, \mathbf{u}_k, t_T)$

$f(\mathbf{u}_{k+1} \,|\, \mathbf{u}_k, t_T)$ is calculated using the GMM model trained in Sec. 3.2 as follows

$$f(\mathbf{u}_{k+1} \,|\, \mathbf{u}_k, t_T) = \sum_{i=i}^{Q}\lambda_i^{(3)}(\gamma_k, \mathbf{u}_k)\phi([\gamma_k, \mathbf{u}_k], \boldsymbol{\mu}_i^{(3)}, \boldsymbol{\Sigma}_i^{(3)}), \tag{37}$$

where γ_k is given by

$$\gamma_k = f_\theta(\Phi^{-1}(F_T(t_T - (l-k) \times w))), \tag{38}$$

And the elements in Eq. (34) are calculated with the parameters from the whole GMM using

$$\boldsymbol{\mu}_i^{(3)} = \boldsymbol{\mu}_{i,2} + \boldsymbol{\Sigma}_{i,20}\boldsymbol{\Sigma}_{i,00}^{-1}([\gamma_k, \mathbf{u}_k] - \boldsymbol{\mu}_{i,0}), \tag{39}$$

$$\boldsymbol{\Sigma}_i^{(3)} = \boldsymbol{\Sigma}_{i,22} - \boldsymbol{\Sigma}_{i,20}\boldsymbol{\Sigma}_{i,00}^{-1}\boldsymbol{\Sigma}_{i,02}, \tag{40}$$

$$\lambda_i^{(3)}(\gamma_k, \mathbf{u}_k) = \frac{\lambda_i \phi([\gamma_k, \mathbf{u}_k], \boldsymbol{\mu}_{i,0}, \boldsymbol{\Sigma}_{i,00})}{\sum_{q=1}^{Q} \lambda_k \phi([\gamma_k, \mathbf{u}_k], \boldsymbol{\mu}_{q,0}, \boldsymbol{\Sigma}_{q,00})}, \tag{41}$$

With the prior distribution and likelihood function, we will be able to calculate the posterior distribution of RUL using Bayesian inference and GMM.

E. Summary

Fig. 7 summarizes the overall procedure of RUL prediction using GMM. The prior distribution and the likelihood function for the first step and the rest are different. For the first step, the prior distribution is calculated from the first part of conditional GMM while the likelihood is calculated from the second part of conditional GMM. For the rest steps, the prior distribution is calculated by combining the posterior distribution from the last two steps, noticed that at the second step, the posterior distribution is used twice. The likelihood is calculated from the whole conditional GMM. Then the posterior distribution can be calculated by applying Bayesian inference using Eq. (22).

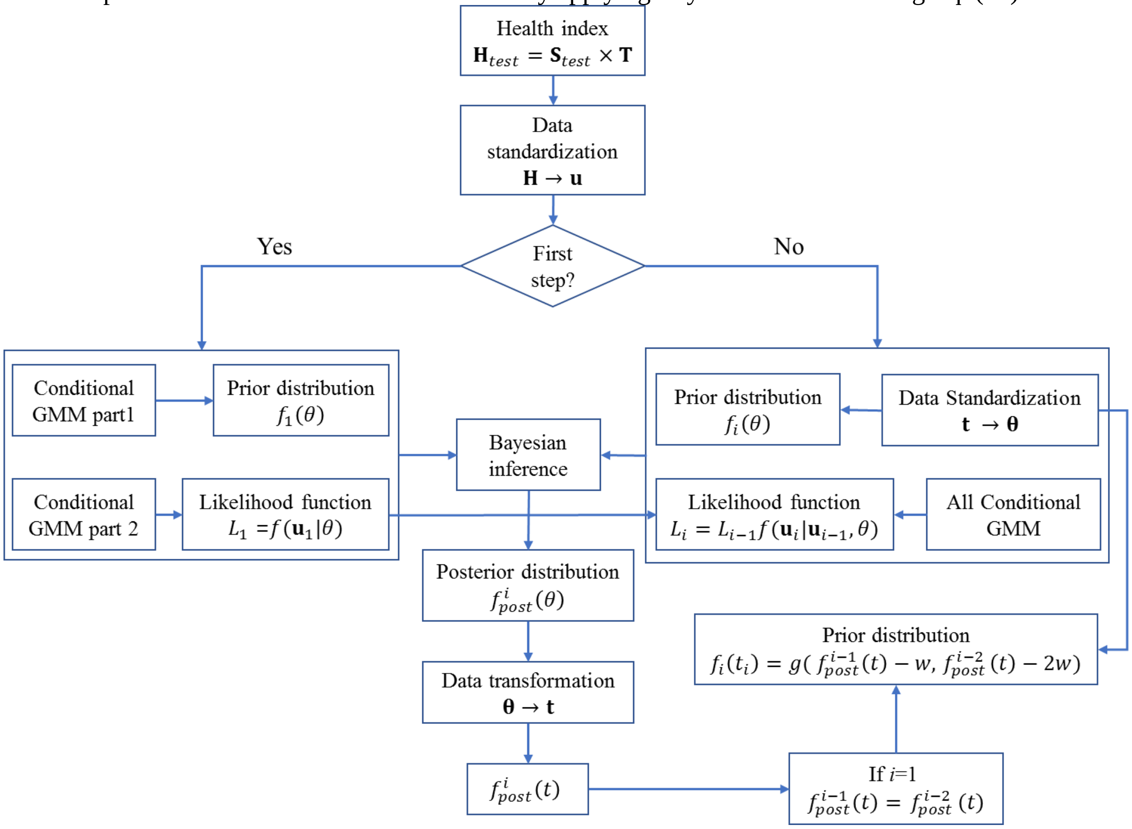

Figure.7 Flowchart of the online prediction phase of multi-GMM based approach

IV. Case Study

In this section, an engineering application problem and the PHM08 Prognostics Data Challenge Dataset are used to demonstrate the effectiveness of the proposed approaches.

A. An engineering problem

The dataset of a civil structure collected from 46 sensors are used as the first example to demonstrate the effectiveness of the proposed method. It consists 120 groups of data of 46 sensors in total. 100 of them are used as training set while the remaining 20 of them are employed as testing set. In these datasets, the first column represents the running time of the system. The second column is the response of the system. If the response is greater than 100, the system fails. The rest columns are the 46 sensor readings.

The score of the performance assessment for one testing set is computed using the following function [20]

$$d = RUL_{predict} - RUL_{true}$$
$$S = \begin{cases} e^{-d/13} - 1, d \le 0 \\ e^{-d/10} - 1, d > 0 \end{cases} \tag{42}$$

The lower the score is, the more accurate the predictions are.

Using GMM-based approach with step size *s*=2 and window size *w*=8, the RUL estimation update of every window is shown in Fig. 8. The score of each testing set is shown in Fig. 9 and the average score using Eq. (41) is 0.68.

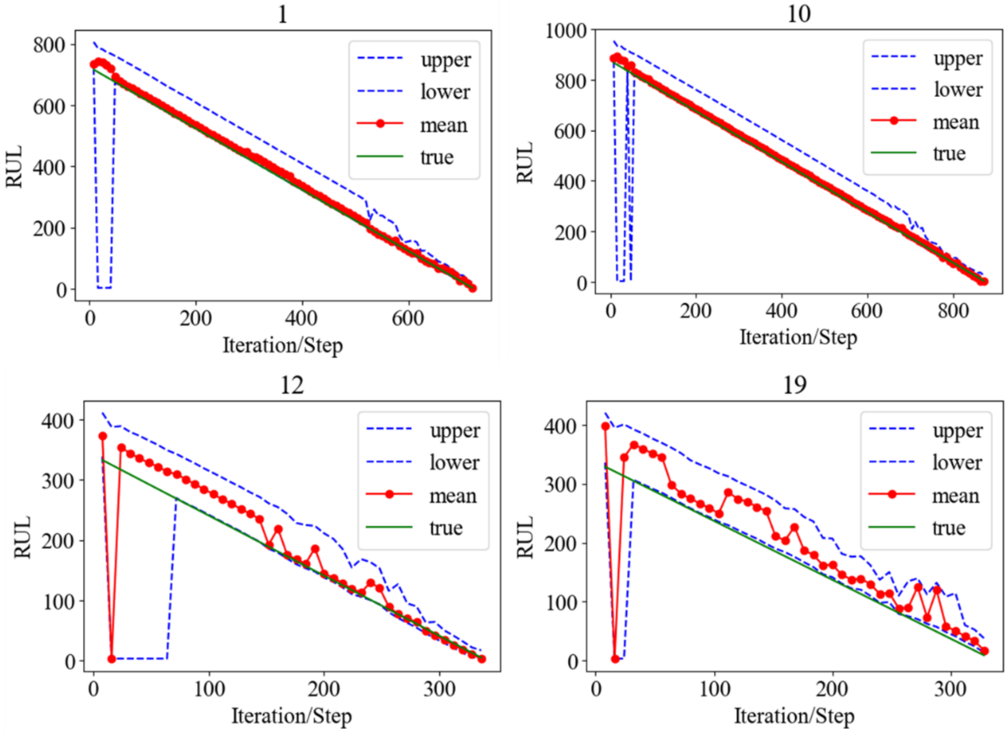

Figure.8 Examples of RUL updating for the engineering problem

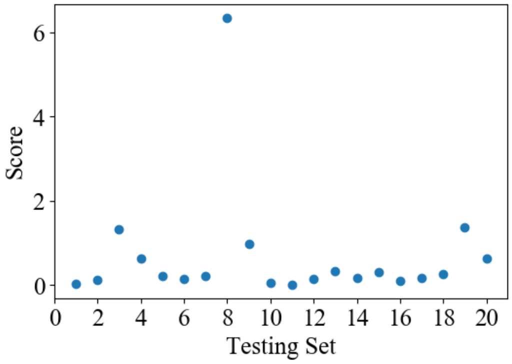

Figure.9 Score of each testing set

B. PHM08 Prognostics Data Challenge Dataset

This dataset has been widely used to test the performance of RUL estimation approaches. The data consists 218 training data sets and 218 testing sets. For each data set, there are 26 columns. The first column represents the unit ID. The second column is the time in cycle. The 3rd to 5th columns are the operation settings and the rest columns are the 46 sensor readings. More details about the PHM08 prognostics data challenge dataset is available in Ref. [26].

Using GMM-based approach with step size *s*=2 and window size *w*=8, the RUL estimation update of every window is shown in Fig. 10. The true and predicted RUL of each testing set is shown in Fig. 11 and the average score using Eq. (41) is 15.62. Table. 1 compares the score of our approach to the state-of-art approaches using the score from Ref. [30]. From the result we can tell that our approach can achieve similar accuracy with the state-of-art approaches.

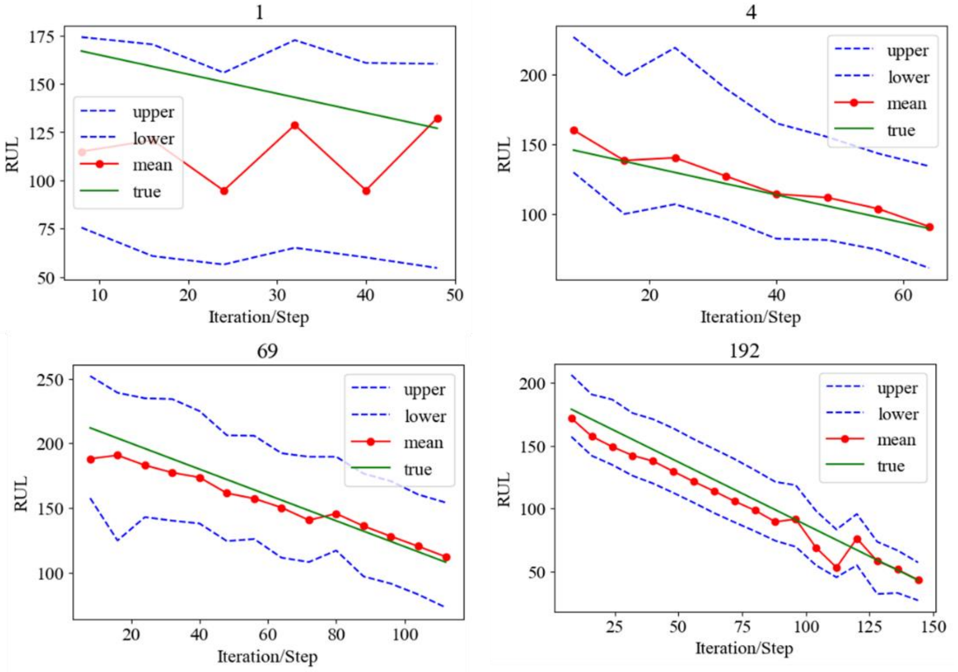

Figure.10 Examples of RUL updating for PHM08 challenge dataset

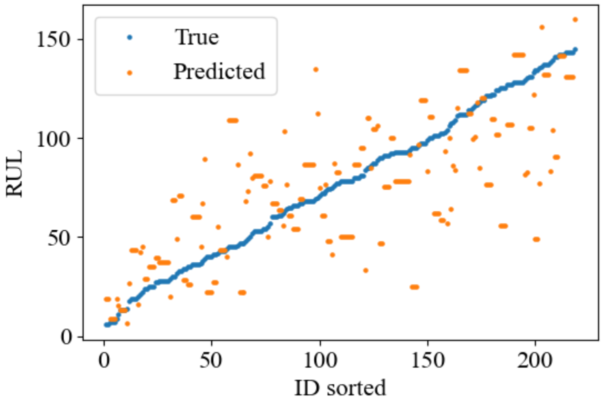

Figure.11 True and predicted RUL comparison

Table .1 Comparison of score for RUL prediction

Approach	Ave. Score
SBI-RVM	10.24
SBI-SVM	9.39
SBI-LSE	10.47
BLR-Quad	247.01
RNN	20.15
Ensemble-AW	8.55
GMM-based approach	15.62

V. Conclusion and Future Work

We have developed a RUL estimation approaches, called GMM-based approach. We compared our approach with similarity-based approach. Using two engineering problems and PHM08 data set, it is shown that our approach has good accuracy. However, we still used the traditional way of data preprocessing to transfer sensor readings into health index, which will reduce the accuracy of our prediction. In the future, we plan to establish a model that can directly connect the sensor reading with remaining useful life and without any other assumptions.

References

1. Banjevic, D. "Remaining useful life in theory and practice," *Metrika* Vol. 69, No. 2-3, 2009, pp. 337-349.
2. Brown, D. W., Kalgren, P. W., Byington, C. S., and Roemer, M. J. "Electronic prognostics–a case study using global positioning system (GPS)," *Microelectronics Reliability* Vol. 47, No. 12, 2007, pp. 1874-1881.
3. Loutas, T. H., Roulias, D., and Georgoulas, G. "Remaining useful life estimation in rolling bearings utilizing data-driven probabilistic e-support vectors regression," *IEEE Transactions on Reliability* Vol. 62, No. 4, 2013, pp. 821-832.
4. Saxena, A., Goebel, K., Simon, D., and Eklund, N. "Damage propagation modeling for aircraft engine run-to-failure simulation," *2008 international conference on prognostics and health management*. IEEE, 2008, pp. 1-9.
5. Abu-Hanna, A., and Lucas, P. J. "Prognostic models in medicine," *Methods of information in medicine* Vol. 40, No. 01, 2001, pp. 1-5.
6. Hyndman, R. J., and Koehler, A. B. "Another look at measures of forecast accuracy," *International journal of forecasting* Vol. 22, No. 4, 2006, pp. 679-688.
7. Si, X.-S., Wang, W., Hu, C.-H., and Zhou, D.-H. "Remaining useful life estimation–a review on the statistical data driven approaches," *European journal of operational research* Vol. 213, No. 1, 2011, pp. 1-14.
8. Carr, M., and Wang, W. "A case comparison of a proportional hazards model and a stochastic filter for condition-based maintenance applications using oil-based condition monitoring information," *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability* Vol. 222, No. 1, 2008, pp. 47-55.
9. Cui, L., Loh, H. T., and Xie, M. "Sequential inspection strategy for multiple systems under availability requirement," *European Journal of Operational Research* Vol. 155, No. 1, 2004, pp. 170-177.
10. Elwany, A. H., and Gebraeel, N. Z. "Sensor-driven prognostic models for equipment replacement and spare parts inventory," *IIe Transactions* Vol. 40, No. 7, 2008, pp. 629-639.
11. Jardine, A. K., Lin, D., and Banjevic, D. "A review on machinery diagnostics and prognostics implementing condition-based maintenance," *Mechanical systems and signal processing* Vol. 20, No. 7, 2006, pp. 1483-1510.
12. Le Son, K., Fouladirad, M., Barros, A., Levrat, E., and Iung, B. "Remaining useful life estimation based on stochastic deterioration models: A comparative study," *Reliability Engineering & System Safety* Vol. 112, 2013, pp. 165-175.
13. Cui, L., Wang, X., Xu, Y., Jiang, H., and Zhou, J. "A novel switching unscented Kalman filter method for remaining useful life prediction of rolling bearing," *Measurement* Vol. 135, 2019, pp. 678-684.
14. Xi, Z., Jing, R., Wang, P., and Hu, C. "A copula-based sampling method for data-driven prognostics," *Reliability Engineering & System Safety* Vol. 132, 2014, pp. 72-82.
15. Si, X.-S., Wang, W., Hu, C.-H., Chen, M.-Y., and Zhou, D.-H. "A Wiener-process-based degradation model with a recursive filter algorithm for remaining useful life estimation," *Mechanical Systems and Signal Processing* Vol. 35, No. 1-2, 2013, pp. 219-237.
16. Heimes, F. O. "Recurrent neural networks for remaining useful life estimation," *2008 international conference on prognostics and health management*. IEEE, 2008, pp. 1-6.
17. Li, X., Ding, Q., and Sun, J.-Q. "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering & System Safety* Vol. 172, 2018, pp. 1-11.
18. Zhang, C., Lim, P., Qin, A. K., and Tan, K. C. "Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics," *IEEE transactions on neural networks and learning systems* Vol. 28, No. 10, 2016, pp. 2306-2318.

19. Babu, G. S., Zhao, P., and Li, X.-L. "Deep convolutional neural network based regression approach for estimation of remaining useful life," *International conference on database systems for advanced applications*. Springer, 2016, pp. 214-228.
20. Wang, T., Yu, J., Siegel, D., and Lee, J. "A similarity-based prognostics approach for remaining useful life estimation of engineered systems," *2008 international conference on prognostics and health management*. IEEE, 2008, pp. 1-6.
21. Reynolds, D. A. "Gaussian Mixture Models," *Encyclopedia of biometrics* Vol. 741, 2009.
22. Povey, D., Burget, L., Agarwal, M., Akyazi, P., Kai, F., Ghoshal, A., Glembek, O., Goel, N., Karafiát, M., and Rastrow, A. "The subspace Gaussian mixture model—A structured model for speech recognition," *Computer Speech & Language* Vol. 25, No. 2, 2011, pp. 404-439.
23. Zivkovic, Z. "Improved adaptive Gaussian mixture model for background subtraction," *Proceedings of the 17th International Conference on Pattern Recognition, 2004. ICPR 2004.* Vol. 2, IEEE, 2004, pp. 28-31.
24. Hu, Z., and Mahadevan, S. "Probability models for data-driven global sensitivity analysis," *Reliability Engineering & System Safety* Vol. 187, 2019, pp. 40-57.
25. Dempster, A. P., Laird, N. M., and Rubin, D. B. "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society: Series B (Methodological)* Vol. 39, No. 1, 1977, pp. 1-22.
26. Akaike, H. "Information theory as an extension of the maximum likelihood principle. Á In: Petrov, BN and Csaki, F," *Second International Symposium on Information Theory. Akademiai Kiado, Budapest, pp. 276Á281*. 1973.
27. Schwarz, G. "Estimating the dimension of a model," *The annals of statistics* Vol. 6, No. 2, 1978, pp. 461-464.
28. Akaike, H. "A new look at the statistical model identification," *IEEE transactions on automatic control* Vol. 19, No. 6, 1974, pp. 716-723.
29. Wit, E., Heuvel, E. v. d., and Romeijn, J. W. "'All models are wrong...': an introduction to model uncertainty," *Statistica Neerlandica* Vol. 66, No. 3, 2012, pp. 217-236.
30. Hu, C., Youn, B. D., Wang, P., and Yoon, J. T. "Ensemble of data-driven prognostic algorithms for robust prediction of remaining useful life," *Reliability Engineering & System Safety* Vol. 103, 2012, pp. 120-135.