HAL
open science

Weighted-feature and cost-sensitive regression model for component continuous degradation assessment

Jie Liu, Enrico Zio

To cite this version:

Jie Liu, Enrico Zio. Weighted-feature and cost-sensitive regression model for component continuous degradation assessment. Reliability Engineering and System Safety, 2017, 168, pp.210 - 217. 10.1016/j.ress.2017.03.012 . hal-01652194

HAL Id: hal-01652194

https://hal.science/hal-01652194

Submitted on 30 Nov 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Reliability Engineering and System Safety 168 (2017) 210–217

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress

Weighted-feature and cost-sensitive regression model for component continuous degradation assessment

Jie Liu [a], Enrico Zio [a,b,*]

[a] Chair on System Science and the Energetic Challenge, EDF Foundation, Laboratoire Genie Industriel, CentraleSupélec, Université Paris-Saclay, Grande voie des Vignes, 92290 Chatenay-Malabry, France

[b] Energy Department, Politecnico di Milano, Milano, Italy

ARTICLE INFO

Keywords:
Condition-based maintenance
Cost-sensitive
Continuous degradation assessment
Feature vector regression
Feature vector selection
Weighted-feature

ABSTRACT

Conventional data-driven models for component degradation assessment try to minimize the average estimation accuracy on the entire available dataset. However, an imbalance may exist among different degradation states, because of the specific data size and/or the interest of the practitioners on the different degradation states. Specifically, reliable equipment may experience long periods in low-level degradation states and small times in high-level ones. Then, the conventional trained models may result in overfitting the low-level degradation states, as their data sizes overwhelm the high-level degradation states. In practice, it is usually more interesting to have accurate results on the high-level degradation states, as they are closer to the equipment failure. Thus, during the training of a data-driven model, larger error costs should be assigned to data points with high-level degradation states when the training objective minimizes the total costs on the training dataset. In this paper, an efficient method is proposed for calculating the costs for continuous degradation data. Considering the different influence of the features on the output, a weighted-feature strategy is integrated for the development of the data-driven model. Real data of leakage of a reactor coolant pump is used to illustrate the application and effectiveness of the proposed approach.

© 2017 Elsevier Ltd. All rights reserved.

1. Introduction

Condition-Based Maintenance (CBM) has gained much attention recently [23,32,8]. Advanced sensors implemented in production systems measure physical variables related to the equipment degradation and proper maintenance is recommended based on the assessed degradation of the equipment. Compared to corrective and scheduled maintenance, CBM can reduce the direct and indirect costs of maintenance and prevent catastrophic failure [1].

One of the cornerstones of CBM is the precise assessment of the current degradation state of the equipment of interest. If it is monitored directly by sensors, it is relatively easy to identify the system degradation state. Otherwise, the degradation state of the equipment of interest needs to be informed from the related monitored variables. For the latter case, physical [13,25], knowledge-based [12,26] or data-driven models [16,33] can be integrated depending on the available knowledge, information and data on the degradation [36,46]. Data-driven models, in particular, benefit from high computation speed and advanced sensors. These models can be classified also based on the type of the underlying degradation process, i.e. proportional hazard models [30], discrete-state degradation [24], continuous-state degradation [1].

In this work, we focus on the data-driven models with supervised learning methods for continuous degradation assessment. Assuming that a pool of degradation patterns of similar equipment are available, a data-driven model is trained off-line with the recorded degradation data and, then, it is used to assess on-line the degradation state of an equipment under operation. Shen et al. [34] adopt a fuzzy support vector data description to construct a fuzzy-monitoring coefficient, which serves as a monotonic damage index of bearing degradation. Logistic regression models and incremental rough support vector data description are used separately in Caesarendra et al. [6] and Zhu et al. [45] for bearing degradation assessment. Principal component analysis is used in Gómez et al. [10] to obtain soil degradation indexes for distinguishing between olive farms with low soil degradation and those in a serious state of degradation. Peng et al. [28] use an inverse Gaussian process model for degradation modelling. Nonhomogeneous continuous-time hidden semi-Markov process is proposed in Moghaddass and Zuo [24] for modelling the degradation of devices undergoing discrete, multistate degradation. Vale and Lurdes [37] employ a stochastic model for character-

* Corresponding author at: Chair on System Science and the Energetic Challenge, EDF Foundation, Laboratoire Genie Industriel, CentraleSupélec, Université Paris-Saclay, Grande voie des Vignes, 92290 Chatenay-Malabry, France.

E-mail address: enrico.zio@polimi.it (E. Zio).

http://dx.doi.org/10.1016/j.ress.2017.03.012
Available online 22 March 2017
0951-8320/© 2017 Elsevier Ltd. All rights reserved.

izing the track geometry degradation process in the Portuguese railway Northern Line.

For the development of the previous data-driven models, it is assumed that the pool of available degradation data is large and representative of the different degradation states of the equipment. However, in practical cases, especially for highly reliable equipment, a problem of imbalance often exists between different degradation states with respect to the knowledge, information and data available to characterize them. For highly reliable equipment, typically there is a long period without degradation or with low-level degradation states, and the data representative of high-level degradation states are relatively limited. Training a data-driven model on the entire set of recorded data with the objective of minimizing the average estimation error on the whole degradation process may lead to a relatively worse performance of the trained model for the data of high level degradation states, as the data on the low-level degradation states outnumber those on the high-level degradation states and the trained model overfits on the low-level degradation states. On the other hand, if the data on the high-level degradation states is not sufficiently large and representative, training a data-driven model only with the data of high-level degradation states may lead to overfitting the recorded data and low generalizability on test data. Furthermore, even if the data on different degradation states are comparable and representative, the industrial practitioner may be more interested in some degradation states, e.g. the high-level degradation states, the peak values during the degradation process, as they are more critical for the functioning of the equipment and, thus, require more accuracy. So, the data-driven model should be trained on the whole dataset, but the objective can not be that of minimizing the average estimation error on the whole dataset. In this work, an approach is proposed based on cost-sensitive regression models in combination of a weighted-feature strategy for the imbalance problem in continuous degradation assessment. To the authors' knowledge, no work has been reported on this problem for continuous degradation assessment.

Cost-sensitive models are very popular for solving classification problems with different costs for different misclassification errors [9]. They have been successfully applied in medical diagnosis [39], object detection [5], intrusion detection [15], face recognition [42], software defect prediction [17] etc. The objective of training a cost-sensitive model is to minimize the total cost on the misclassification error in a way to assign a larger cost to the error on the minority class than that on the majority class. Cost-sensitive models have been integrated in neural networks [14], decision trees [41], support vector machines [38], Bayesian networks [11], etc. Different methods can be used to assign the costs for different misclassification errors and the most popular one is to set the costs based on expertise on the problem [40]. Different cost-assignment methods can be proposed considering the specific requirements and characteristics of the application. In the mentioned previous works, discrete costs are assigned to different classes, assuming that the cost of the misclassification errors on the data of the same class is the same. This is reasonable for classification problems with finite classes. On the contrary, for regression of continuous degradation, it is infeasible to assign discrete costs for infinite degradation states. One way is to discretize the continuous degradation states, but this may cause loss of information on the degradation states. This work tries to propose a method for assigning continuous costs to degradation states. The proposed method assigns larger costs to the regression errors on the high-level degradation states than those to the low-level degradation states and also larger costs to peak values than to normal values in their neighborhoods.

The basic data-driven model used in this paper for integrating the cost-sensitive strategy is Feature Vector Regression (FVR), a kernel method proposed in Liu and Zio [20]. Training a FVR model requires feature vector selection and regression model construction. Feature vector selection proceeds by selecting part of the training data points as feature vectors in the Reproducing Kernel Hilbert Space (RKHS), where the mapping of all training data points can be expressed as a linear combination of the selected feature vectors. Different to the traditional kernel methods that define the estimation function as a kernel expansion on all the training dataset, the estimate function of FVR is a kernel expansion only on the feature vectors. The objective is to minimize the regression error on the whole training dataset. The parameters in FVR can be calculated analytically, without using a sophisticated method for tuning parameters in the model. FVR model is combined with cost-sensitive and weighted-feature strategies in this work. These strategies are not suitable only for FVR, but also for other regression models for degradation assessment.

Another original contribution of this work is the adopted weighted-feature strategy. Conventionally, after the raw data are pre-processed, some features are selected as inputs and they form directly the input vectors which are used for training a supervised data-driven model. However, the selected features may still have different levels of influence on the output. In order to characterize this difference in kernel methods, larger weights are assigned to the features with a higher influence on the output in the kernel function. Weighted-feature strategy has been used in Amutha and Kavitha [3], Peng et al. [29], Phan et al. [31], Liu et al. [18]. The weighted-feature strategy is integrated for the first time with the basic model (FVR) in this work and it is also the first time that the weighted-feature strategy is used for degradation assessment. An efficient optimization process is adopted in this work for finding the weights for different features and the parameters in FVR.

To demonstrate the proposed approach, we make use of a case study concerning the estimation of leakage of coolant water from the seal of a Reactor Coolant Pump (RCP) in a Nuclear Power Plant (NPP). The control of the leakage is very important for safety reasons [21]. Sensors are installed to monitor the temperature, pressure, flow of coolant water. These variables are informative for assessing the leakage amount. When the amount of leakage is low, the reactor can make up the leakage by using auxiliary and protection systems. But when it is large, the NPP must be shut down. Thus, operators are more concerned of correctly identifying the leakages of large magnitude than small. Correspondingly, the accuracy is more important in case of large amount of leakage than of small amount. However, as the NPP components are highly reliable, most of the historical recorded data relates to low amounts of leakage, the data on large amounts of leakage being far less. In this work, the proposed weighted-feature and cost-sensitive FVR is adopted for estimating continuously the leakage from the RCP.

The remainder of the paper is structured as follows. Section 2 recalls briefly the FVR method and introduces the approach proposed for assessing continuous degradation with imbalance. The case study on the leakage from RCP is presented in Section 3 with experimental results. Some conclusions and perspectives are drawn in Section 4.

2. Weighted-feature and cost-sensitive FVR for degradation assessment

FVR proposed in Liu and Zio [20] is a kernel method, which allows easy tuning of hyperparameters and offers a geometrical interpretation. FVR reduces the size of the estimate function by selecting a number of feature vectors and keeps its generalization power by minimizing the error on the whole training dataset. In this Section, FVR is briefly reviewed firstly and, then, the weighted-feature and cost-sensitive FVR is introduced, with details on the calculation of the costs and the tuning of the weight of each feature.

In the paper, $(\boldsymbol{x}, y)$ represents the input and corresponding output of a nonlinear relation. $k(\boldsymbol{x}_1, \boldsymbol{x}_2)$ is the kernel function, taken as the inner product of $\varphi(\boldsymbol{x}_1)$ and $\varphi(\boldsymbol{x}_2)$, i.e. $\langle \varphi(\boldsymbol{x}_1), \varphi(\boldsymbol{x}_2) \rangle$, and $\varphi(\boldsymbol{x})$ is the function that maps the original data into a high dimensional space (i.e. Reproducing Kernel Hilbert Space (RKHS)), where the relation between input and output becomes linear. The training dataset is $\mathbf{T} = \{(\boldsymbol{x}_i, y_i)\}$, for $i = 1, 2, \ldots, N$. Without restricting the generalizability of the model,

Initialization:
Training dataset: $\mathbf{T} = \{(\boldsymbol{x}_i, y_i)\}$, for $i = 1, 2, \ldots, N$
Feature space: $\mathbf{S} = [\,]$
Threshold of LF: τ
FVS:
First FV in $\mathbf{S}$:
For $i = 1$ to N calculate
$\boldsymbol{S} = \{\boldsymbol{x}_i\}$, compute the sum of the $LF(\boldsymbol{x})$ for all training data points with respect to the present $\boldsymbol{S}$.
End for.
Select the point which gives the minimum of the sum of the $LF(\boldsymbol{x})$ as the first feature vector and add it to $\mathbf{S}$ as the first feature vector. $\mathbf{T}$ is reduced to the complement of $\mathbf{S}$ in $\mathbf{T}$, i.e. $\mathbf{T} = \mathbf{T}\backslash\mathbf{S}$.
Second and following feature vectors:
Calculate the local fitness for all data points in $\mathbf{T}$ with respect to the present feature space $\mathbf{S}$;
Select the data point k which gives the maximum local fitness;
If $LF(\boldsymbol{x}_k) > \tau$, this point is a new feature vector and added to $\mathbf{S}$; and $\mathbf{T} = \mathbf{T}\backslash\mathbf{E}$, with $\mathbf{E}=\{(\boldsymbol{x}_i, y_i)$: $LF(\boldsymbol{x}_i) \le \tau$ and $(\boldsymbol{x}_k, y_k)\}$;
If $LF(\boldsymbol{x}_k) \le \tau$, end the process of feature vector selection.

Fig. 1. Pseudo-code of the feature vector selection process.

we assume that high-level degradation states correspond to high values of y.

2.1. Feature vector regression

Schölkopf et al. [35] propose nonparametric and semi-parametric representer theorems demonstrating that kernel algorithms with the minimal sum of an empirical risk term and a regularization term in RKHS as optimization objective achieve their optimal solutions for an estimate function that is a kernel expansion on the training data points. Specifically, in mathematical terms, such estimate function $f(\boldsymbol{x})$ can be expressed as

$$f(\boldsymbol{x}) = \sum_{i=1}^{N} \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}) + b = \sum_{i=1}^{N} \alpha_i \langle \varphi(\boldsymbol{x}_i), \quad \varphi(\boldsymbol{x})\rangle + b, \tag{1}$$

where α_i, $i = 1, 2, \ldots, \quad N$ are the Lagrange multipliers and b is a constant.

If there exists such a subset $\boldsymbol{S} = \{(\boldsymbol{x}_i^s, y_i^s)\}, i = 1, 2, \ldots, M$, with $M < N$, such that the mapping of each training data point can be expressed as a linear combination of the mapping of $\boldsymbol{S}$ RKHS, i.e. $\boldsymbol{\varphi}(\boldsymbol{x}) = \sum_{i=1}^{M} \beta_i(\boldsymbol{x})\boldsymbol{\varphi}(\boldsymbol{x}_i^s)$, after replacing $\varphi(\boldsymbol{x})$ in Eq. (1) with $\sum_{i=1}^{M} \beta_i(\boldsymbol{x})\boldsymbol{\varphi}(\boldsymbol{x}_i^s)$, the estimate function $f(\boldsymbol{x})$ can be rewritten as

$$f(\boldsymbol{x}) = \sum_{i=1}^{N}\sum_{j=1}^{M} \alpha_i \beta_j(\boldsymbol{x}) \langle \boldsymbol{\varphi}(\boldsymbol{x}_i), \boldsymbol{\varphi}(\boldsymbol{x}_j^s)\rangle + b = \sum_{j=1}^{M} \beta_j(\boldsymbol{x}) \left(\sum_{i=1}^{N} \alpha_i \langle \boldsymbol{\varphi}(\boldsymbol{x}_i), \boldsymbol{\varphi}(\boldsymbol{x}_j^s)\rangle\right) + b;$$

thus,

$$f(\boldsymbol{x}) = \sum_{i=1}^{M} \beta_i(\boldsymbol{x})(\hat{y}_i^s - b) + b, \tag{2}$$

with $\hat{y}_i^s = f\left(\boldsymbol{x}_i^s\right) = \sum_{i=1}^{N} \alpha_i k(\boldsymbol{x}_i, \boldsymbol{x}_i^s) + b.$

With Eq. (2), FVR formulates the optimization problem as

$$\begin{aligned} &\textit{Minimize}_{\ \hat{y}_j^s, b} W = \frac{1}{N}\sum_{i=1}^{N}\left(f\left(\boldsymbol{x}_i\right) - y_i\right)^2 \\ &\textit{Subject to } f\left(\boldsymbol{x}_i\right) = \sum_{j=1}^{M} \beta_j(\boldsymbol{x}_i)(\hat{y}_j^s - b) + b \end{aligned}, \tag{3}$$

where the unknown parameters (decision variables) to optimize are $\hat{y}_j^s$, $j = 1, 2, \ldots, M$ and b, and the objective of the optimization problem is the minimal estimation error on the whole training dataset.

By setting the partial derivatives of W with respect to $\hat{y}_j^s$ and b to zero, the previous parameters can be calculated by solving the following system of equations:

$$\begin{bmatrix} \boldsymbol{\Omega} & \mathbf{H} \\ \boldsymbol{\Gamma}^T & c \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{y}}^s \\ b \end{bmatrix} = \begin{bmatrix} \mathbf{P} \\ l \end{bmatrix}, \tag{4}$$

where $\boldsymbol{\Omega}$ is a $M \times M$ matrix with $\boldsymbol{\Omega}_{mn} = \sum_{i=1}^{N} \beta_m(\boldsymbol{x}_i) * \beta_n(\boldsymbol{x}_i)$, $\mathbf{H}$ is a $M \times 1$ vector with $\mathbf{H}_m = \sum_{i=1}^{N} \beta_m(\boldsymbol{x}_i) * (1 - \sum_{j=1}^{M} \beta_j(\boldsymbol{x}_i))$, $\boldsymbol{\Gamma}$ is a $M \times 1$ vector with $\boldsymbol{\Gamma}_m = \sum_{i=1}^{N} \beta_m(\boldsymbol{x}_i) * (1 - \sum_{l=1}^{M} \beta_l(\boldsymbol{x}_i))$, c is a constant and $c = \sum_{i=1}^{N} (1 - \sum_{j=1}^{M} \beta_j(\boldsymbol{x}_i))^2$; $\hat{\boldsymbol{y}}^s = (\hat{y}_j^s), \quad j = 1, 2, \ldots, \quad M$ and b are the unknowns in Eq. (3), $\mathbf{P}$ is a $M \times 1$ vector with $\mathbf{P}_m = \sum_{i=1}^{N} \beta_m(\boldsymbol{x}_i) * y_i$, $l = \sum_{i=1}^{N}(1 - \sum_{j=1}^{M} \beta_j(\boldsymbol{x}_i)) * y_i$.

Feature Vector Selection (FVS) method proposed in Baudat and Anouar [4] can find the previous subset $\boldsymbol{S}$. The main idea is to find an oblique coordinate system in the RKHS of $k(\boldsymbol{x}_1, \boldsymbol{x}_2)$ and, then, each vector in RKHS can be expressed as a linear combination of the coordinate vectors. The FVS method can build such an oblique coordinate system with the mapping of a number of the training data points. Experimental results show that the number of the selected data points may be far less than that of the training dataset, i.e. $M \ll N$. A modified version of the FVS method is proposed in Liu and Zio [20] to speed the selection process. The local fitness of one data point $\boldsymbol{x}$with respect to the current $\boldsymbol{S}$ is calculated as $LF(\boldsymbol{x}) = |1 - K_{\boldsymbol{S},x}^t K_{\boldsymbol{S},\boldsymbol{S}}^{-1} K_{\boldsymbol{S},x}/(k(\boldsymbol{x}, \boldsymbol{x}))|$. Details on the method can be found in Baudat and Anouar [4] and Liu and Zio [20]. For completeness, the pseudo-code of the feature vector selection process proposed in Liu and Zio [20] is shown in Fig. 1.

The parameters $\boldsymbol{\beta}(\boldsymbol{x}) = \{\beta_i(\boldsymbol{x})\}, i = 1, 2, \ldots, M$ in Eqs. (2)–(4) can be calculated with the selected feature vectors in $\boldsymbol{S}$ as in Eq. (5):

$$\boldsymbol{\beta}(\boldsymbol{x}) = \quad K_{\boldsymbol{S},x}^t K_{\boldsymbol{S},\boldsymbol{S}}^{-1}, \tag{5}$$

where $K_{s,s}$ is the kernel matrix of $\boldsymbol{S}$ and $K_{\boldsymbol{S},x} = \{k(\boldsymbol{x}_i, \quad \boldsymbol{x})\}$, $i = 1, 2, \ldots, \quad M$.

2.2. Weighted-feature and cost-sensitive FVR

For the situation with data imbalance mentioned in the Introduction, the minimal average error as in Eq. (3) may not give satisfactory results. To avoid this, different costs should be assigned to the errors on different data points during the training process, in order to satisfy the desired accuracy on some specific data points of interest.

Another issue regards the usefulness of different features for the output assessment: the features need to be treated differently and larger weights should be assigned to the more useful ones. In kernel methods, different weights can be assigned to different features by using kernel function $k(\boldsymbol{\omega}.*\boldsymbol{x}_1, \boldsymbol{\omega}.*\boldsymbol{x}_2)$ instead of $k(\boldsymbol{x}_1, \boldsymbol{x}_2)$, where $\boldsymbol{\omega}$ is the weight vector (of the same size as $\boldsymbol{x}$) of the usefulness of each feature and operator.* is the element-wise multiplication.

Initiating the costs of different data points as $\boldsymbol{C} = [c_1, c_2, \dots, c_N]$, the optimization problem in Eq. (3) can be reformulated as

$$\begin{aligned} &\textit{Minimize}_{\hat{y}_j^s, b} W = \frac{1}{N}\sum_{i=1}^{N} c_i\left(f\left(\boldsymbol{x}_i\right) - y_i\right)^2 \\ &\textit{Subject to } f\left(\boldsymbol{x}_i\right) = \sum_{j=1}^{M} \beta_j(\boldsymbol{x}_i)(\hat{y}_j^s - b) + b \end{aligned} \quad . \tag{6}$$

The unknown parameters $\hat{y}_j^s$, $j = 1, 2, \dots, M$ and b in Eq. (6) can be calculated by solving the following equation:

$$\begin{bmatrix} \boldsymbol{\Omega} & \mathbf{H} \\ \boldsymbol{\Gamma}^T & c \end{bmatrix} \begin{bmatrix} \hat{\boldsymbol{y}}^s \\ b \end{bmatrix} = \begin{bmatrix} \mathbf{P} \\ l \end{bmatrix}, \tag{7}$$

where $\boldsymbol{\Omega}$ is a $M \times M$ matrix with $\boldsymbol{\Omega}_{mn} = \sum_{i=1}^{N} c_i \beta_m(\boldsymbol{x}_i) * \beta_n(\boldsymbol{x}_i)$, $\mathbf{H}$ is a $M \times 1$ vector with $\mathbf{H}_m = \sum_{i=1}^{N} c_i \beta_m(\boldsymbol{x}_i) * (1 - \sum_{j=1}^{M} \beta_j(\boldsymbol{x}_i))$, $\boldsymbol{\Gamma}$ is a $M \times 1$ vector with $\boldsymbol{\Gamma}_m = \sum_{i=1}^{N} c_i \beta_m(\boldsymbol{x}_i) * (1 - \sum_{l=1}^{M} \beta_l(\boldsymbol{x}_i))$, c is a constant and $c = \sum_{i=1}^{N} c_i (1 - \sum_{j=1}^{M} \beta_j(\boldsymbol{x}_i))^2$; $\mathbf{P}$ is a $M \times 1$ vector with $\mathbf{P}_m = \sum_{i=1}^{N} c_i \beta_m(\boldsymbol{x}_i) * y_i$, $l = \sum_{i=1}^{N} c_i (1 - \sum_{j=1}^{M} \beta_j(\boldsymbol{x}_i)) * y_i$.

The parameters $\boldsymbol{\beta}(\boldsymbol{x})$ are still calculated with Eq. (5), but with the kernel function $k(\boldsymbol{\omega}.*\boldsymbol{x}_1, \boldsymbol{\omega}.*\boldsymbol{x}_2)$.

In this model, the unknown parameters include the cost vector $\boldsymbol{C} = [c_1, c_2, \dots, c_N]$, the weight vector $\boldsymbol{\omega} = [\omega_1, \omega_2, \dots, \omega_N]$ and the parameters related to the kernel function.

The cost vector $\boldsymbol{C}$ can be calculated analytically. In this work, we focus on the peak values and high-level degradation states. The peak values may occur abruptly and cause the failures of the system. The high-level degradation states are reached close to the failure state of the system and, thus, it is even more important to get good accuracy. Thus, in this work, the cost is dependent on the degradation state and peak value. A large cost is, thus, assigned to data points which are peak values and high-level degradation states. Specifically, for a data point $(\boldsymbol{x}_t, y_t)$ with $t = 1, 2, \dots, N$, its cost is calculated as

$$c_t = e^{(s_t + d_t)/\sigma}, \tag{8}$$

where s_t is the score of the point with respect to the peak function in Eq. (9) [27] and d_t is the normalized degradation value of y_t as in Eq. (10):

$$a_t = \begin{cases} y_t - \frac{1}{2k}\sum_{i=1;\, i\neq t}^{2k+1} y_i, & \textit{if } t \le k \\ y_t - \frac{1}{2k}\sum_{i=t-k;\, i\neq t}^{t+k} y_i, & \textit{if } k < t < N-k, \quad \textit{for } t = 1, \quad 2, \quad \dots, \quad N, \\ y_t - \frac{1}{2k}\sum_{i=N-2k;\, i\neq t}^{N} y_i, & \textit{if } t \ge N-k \end{cases}$$

$$s_t = \frac{a_t - \min(\boldsymbol{a})}{\max(\boldsymbol{a}) - \min(\boldsymbol{a})} * 0.8 + 0.1, \ \textit{with } \boldsymbol{a} = [a_1, \ a_2, \ \dots, \ a_N], \tag{9}$$

$$d_t = \frac{y_t - \min(\boldsymbol{y})}{\max(\boldsymbol{y}) - \min(\boldsymbol{y})} * 0.8 + 0.1, \ \textit{with } \boldsymbol{y} = [y_1, \ y_2, \ \dots, \ y_N]. \tag{10}$$

The cost is composed of two parts: s_t and d_t. Peak degradation values have high scores s_t and high-level degradation states have relatively high values of d_t. The two values, s_t and d_t are normalized in the interval [0.1 0.9] to have equal weights on the cost of one data point. The parameter σ in Eq. (8) is a case-specific value that must be set by the analyst: the smaller the value of σ is, the larger the difference between the different costs.

For the parameters in the kernel function and for the weight vector for the different features, an iterative procedure is proposed (Fig. 2) for minimizing the total cost in Eq. (6). An initial weight vector for the features is set to **1**. With the fixed weight vector, the parameters in the kernel function are tuned and, then, with the tuned parameters value in the kernel function, the weight vector is tuned. The process is repeated until a fixed number of iterations *maxIter* is reached. For tuning the parameters in the kernel function or the weight vector, conventional optimization methods, e.g. genetic algorithm [43], grid search [19], particle swarm optimization [44], ant colony optimization [22] can be used, with the objective of minimizing the total cost on the training dataset as in Eq. (6). Note that during the optimization process, the sum of the weights for the features equals always that of the initial weights. The convergence of the iterative optimization process is shown in Section 3.

3. Case study

The implementation of the proposed approach is sketched in Fig. 3.

In the experiment, the Radial Basis kernel Function (RBF) is used as kernel function, i.e. $k(\boldsymbol{x}_1, \boldsymbol{x}_2) = e^{\frac{-\|\boldsymbol{x}_1 - \boldsymbol{x}_2\|^2}{2\gamma^2}}$. The value of γ can be calculated analytically with Eq. (11) below, as proposed by Cherkassky and Ma [7], whereas the value for μ is chosen between 0 and 1: by trial and error, the value of μ is set to 0.2. Thus, no explicit optimization method, e.g. grid search, is used for tuning the parameters in RBF. Given the weight vector $\boldsymbol{\omega}$, the value of γ is calculated directly with Eq. (11) after replacing $\boldsymbol{x}$ by $\boldsymbol{\omega}.*\boldsymbol{x}$.

$$\gamma^2 = \mu * max\|\boldsymbol{x}_i - \boldsymbol{x}_j\|^2, \quad i, \quad j = 1, \quad \dots, \quad N \tag{11}$$

With the calculated value of γ, a grid search method is used to tune the weights of the different features. Suppose there are K features: the possible relative weight r_i of one feature can be drawn from $R = [1\ 2\ 3\ 4\ 5\ 6\ 7\ 8\ 9]$ with equal probability; for each combination of the relative weights $[r_1, r_2, \dots, r_K]$, the weight vector $\boldsymbol{\omega}$ is calculated as

$$\omega_i = N_f * r_i \Big/ \sum_{j=1}^{K} r_j, \quad i = 1, \quad 2, \quad \dots, \quad K, \tag{12}$$

to satisfy the initial criterion that the sum of the weights equals to the sum of the initial weights for all the features, i.e. N_f which is the number of features. R is an author-specified vector containing the relative weight values used to differentiate the contribution of different input variables to the output.

The number of maximal iterations is set to 15.

3.1. Verification of the proposed approach

In order to verify the effectiveness of the proposed approach, it is tested on several public imbalanced datasets. These imbalanced datasets are taken from Keel data set Repository for binary classification ad regression [2].

The dataset for binary classification present different Imbalance Ratios (IRs). The characteristics of the binary classification datasets are shown in Table 1.

A comparison is carried out between the proposed approach, i.e. weighted-feature and cost-sensitive FVR (WF-CS-FVR), and a standard FVR. Another method is added to the comparison, i.e. cost-sensitive FVR (CS-FVR) without using the weighted feature strategy, i.e. solving Eq. (6) with the kernel function $k(\boldsymbol{x}_1, \boldsymbol{x}_2)$ instead of the weighted-feature one $k(\boldsymbol{\omega}.*\boldsymbol{x}_1, \boldsymbol{\omega}.*\boldsymbol{x}_2)$. The accuracy of the binary classification is characterized by the True Positive Rate (TPR) and True Negative Rate (TNR), which are the percentages of the correctly classified data points in positive and negative classes, respectively.

The results are shown in Table 2. It is seen that by assigning a larger weight to the error on the minority class, the TPR value increases and the TNR value decreases. In the comparison of the results of CS-FVR

Step 1: The weights of all the features are set to one.
Step 2: For iteration = 1 : *maxIter*
Step 2.1: Tune the parameters in the kernel function with fixed feature weights
Step 2.2: Tune the feature weights with the tuned parameters in the kernel function
End For

Fig. 2. Iterative optimization of the parameters in the kernel function and the weight vector.

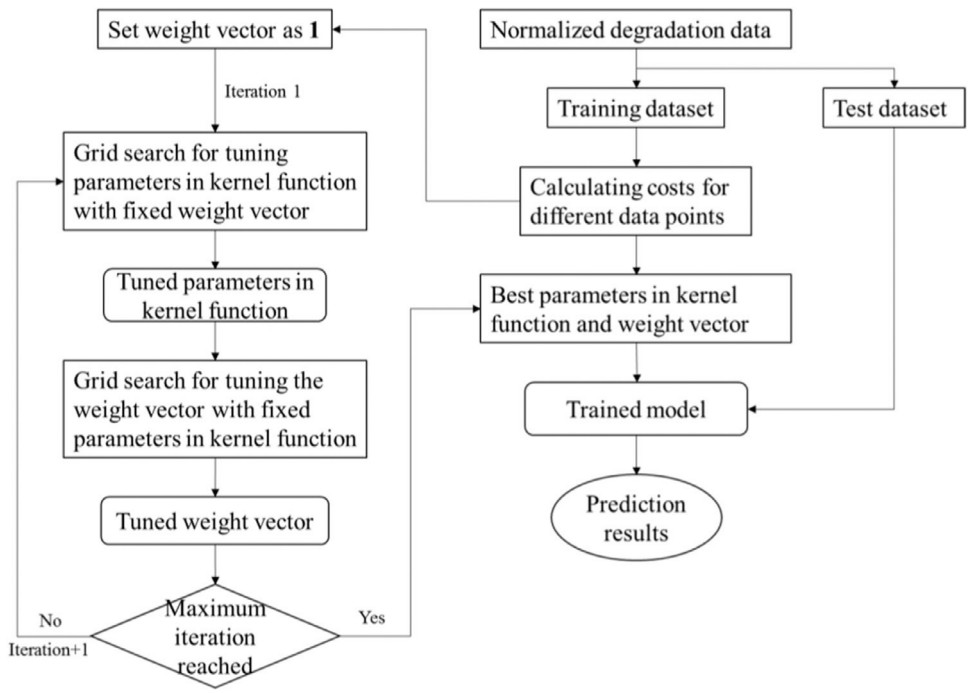

Fig. 3. Proposed approach.

Table 1
Characteristics of the imbalanced binary classification datasets [2].

Dataset	Number of instances	Number of attributes	Imbalance Ratio (IR)
glass1	214	9	1.82
haberman	306	3	2.78
new-thyroid1	215	5	5.14
ecoli3	336	7	8.6
ecoli-0–6-7_vs_5	220	6	10
yeast-1_vs_7	459	7	14.3
ecoli4	336	7	15.8
abalone-9_vs_18	731	8	16.4
shuttle-6_vs_2-3	230	9	22

Table 2
Results of the proposed approach WF-CS-FVR on the public imbalanced datasets for binary classification and comparisons with FVR and CS-FVR.

Datasets	FVR		CS-FVR		WF-CS-FVR	
	TNR	TPR	TNR	TPR	TNR	TPR
glass1	0.6296	0.5625	0.4815	0.6875	0.5926	0.6875
haberman	0.8444	0.1176	0.7556	0.3529	0.7556	0.7059
new-thyroid1	0.9167	0.7143	0.9167	0.7143	0.9722	0.8571
ecoli3	0.9508	0.7143	0.8689	0.8571	0.8689	1.0000
ecoli-0–6-7_vs_5	1.0000	0.5000	1.0000	0.7500	1.0000	0.7500
yeast-1_vs_7	1.0000	0.3333	0.7412	0.5000	0.7882	0.5000
ecoli4	1.0000	0.7500	1.0000	0.7500	1.0000	0.7500
abalone-9_vs_18	0.9928	0.7778	0.8841	1.0000	0.9203	1.0000
shuttle-6_vs_2-3	1.0000	1.0000	1.0000	1.0000	1.0000	1.0000

and WF-CS-FVR, we can observe that by integrating weighted-feature strategies, at least one of the values of TPR and TNR can be improved for most of the datasets. The results of FVR and WF-CS-FVR show that for most of the datasets, by sacrificing the accuracy on the majority class, a relatively large gain can be obtained on the accuracy of the minority class. For example, for dataset 'galss1', the difference of TNR values between FVR and WF-CS-FVR is 0.0370 and the difference of TPR values is 0.1250, which is much bigger than the difference of the TNR values.

Five datasets are used to test the effectiveness of the proposed framework for regression. The characteristics of these datasets are shown in Table 3. The outputs and inputs of these datasets are normalized to [0.0 0.9]. The type of the inputs can be numeric or integer. The data points with output values bigger than 0.6 are higher data points and the others are lower data points. The Mean Squared Errors (MSEs) of the proposed framework and the benchmark methods (i.e. FVR and CS-FVR) on higher and lower data points are shown in Table 4.

From Table 4, the following conclusions can be drawn: 1) the models (CS-FVR and WF-CS-FVR) with cost-sensitive strategy give better results on higher data points; 2) the weighted-feature strategy can further improve the accuracy of the CS-FVR model on the higher data points; 3) the weighted-feature strategy can give better results even on the lower data points than the benchmark methods (e.g. for datasets friedman and concrete) when the input attributes are not well scaled during the

Table 3
Characteristics of the datasets for regression.

Name	# of instances	# of features	Data type
laser	993	4	numeric
autoMPG8	392	7	numeric & integer
forestFires	517	12	numeric & integer
friedman	1200	5	numeric
concrete	1030	8	numeric & integer

Table 4
MSEs of the proposed framework and the benchmark methods on the datasets for regression.

Datasets	FVR		CS-FVR		WF-CS-FVR	
	MSE on higher data points	MSE on lower data points	MSE on higher data points	MSE on lower data points	MSE on higher data points	MSE on lower data points
laser	3.69E−4	8.45E−4	3.30E−4	1.30E−3	1.70E−4	7.12E−4
autoMPG8	2.17E−2	2.20E−3	2.06E−2	2.70E−3	1.90E−3	3.20E−3
forestFires	3.50E−3	2.94E−4	2.30E−3	1.70E−3	2.20E−3	1.80E−3
friedman	1.90E−3	1.50E−2	2.30E−3	1.90E−2	9.10E−4	7.70E−4
concrete	5.02E−4	2.96E−4	2.22E−4	8.40E−4	2.20E−14	9.23E−14

preprocessing part; 4) the accuracy on the lower data points is normally somewhat given up.

3.2. Degradation state estimation in a NPP

The application of the proposed method is shown on a real case concerning the amount of leakage from the seals of an RCP of an NPP.

Twenty variables related to the leakage process are available, e.g. flow of seal injection supply, temperature of the water seal, temperature of seal injection line, temperature of by-pass hot leg loop, pressure in the pressure injection line, etc. The recorded data for eight RCPs from different NPPs are available, with the true leakage and all related variables. The data from the one RCP are used as the test dataset and the data from other seven RCPs are combined in the training dataset. Two case studies from two different NPPs are generated from the data, named Case 1 and Case 2 separately. For confidentiality considerations, the leakage values are normalized in the interval [0.1 0.9]. As shown in Fig. 4 for one RCP, the leakage can take continuous values in the interval [0.1 0.9] and the sizes of the recorded data on different degradation states are very different. Considering a partition of the entire range in different degradation level states, [0.1 0.3], [0.3 0.5], [0.5 0.7], [0.7 0.9], the recorded data with leakage values between [0.1 0.3] are much more numerous than those for leakage values in the interval [0.3 0.5], [0.5 0.7] and [0.7 0.9].

3.2.1. The proposed approach

In the numerical experiment, the costs related to the estimation errors on different data points are calculated by Eq. (8), with the normalized leakage values. The unknown σ in the equation is set to be 0.3 to penalize the importance of the estimation errors on the data points with a leakage lower than 0.3, as shown in Fig. 5 for Case 1. For example,

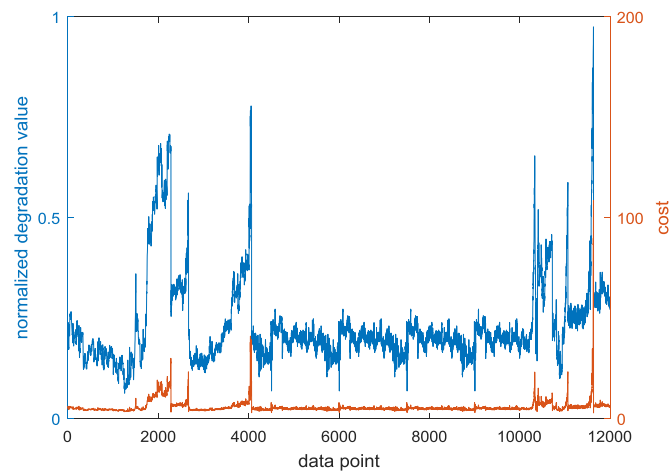

Fig. 5. Costs of different training data points for Case 1.

we can observe that the 1504th data point and the data points numbered between 2400 and 2600 have similar degradation values, but as the 1504th data point is a peak value, its weight (i.e. 10.04) is higher than those of the data points numbered between the 2400 and 2600 (i.e. 5.51–8.01).

3.2.2. Experiment results

Fig. 6 shows the minimal cost of each iteration of the tuning process during the training for Case 1. It is shown that the minimal cost in Eq. (6) converges to a minimal value. The minimal cost is obtained at the 11th iteration and it is stable after the 11th iteration, with small perturbation. The perturbation is caused by the parameter optimizing methods, i.e. analytical method for γ and grid search for $\boldsymbol{\omega}$ which derives only the sub-optimal results.

The comparison results are given in Table 5, with regards to the MSEs. The MSE are calculated separately on the whole test dataset, the test data points with small amount of leakage in [0.1 0.3] (low-level degradation states) and the test data points with large amount of leakage in [0.3 0.9] (high-level degradation states).

From Table 5, one can observe that the MSE of FVR on the low-level degradation states is much smaller than that on the high-level degradation states. The model i.e. FVR is not capable giving satisfactory results on both low-level and high-level degradation states. The MSE values for Case 2 are greater than those for Case 1. This is due to the fact that the

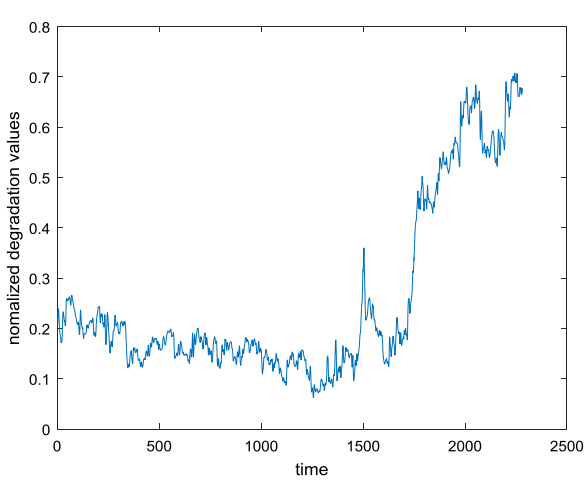

Fig. 4. Normalized leakage values in one RCP.

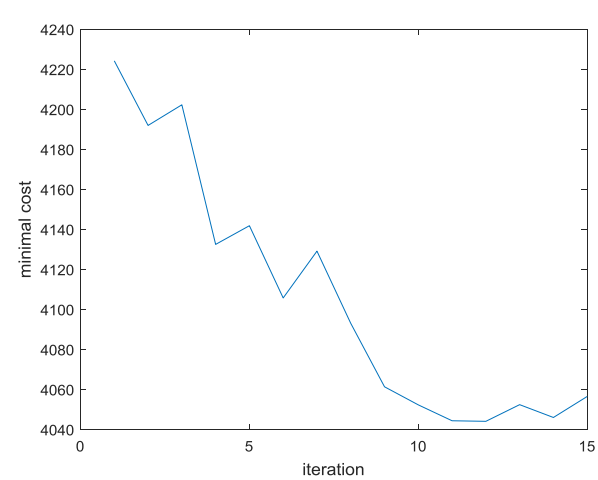

Fig. 6. Convergence of the minimal cost during the training process of Case 1.

Table 5
MSE given by the methods (FVR, CS-FVR, WF-CS-FVR) for the test dataset.

MSE	Whole test dataset		Data with leakage in [0.1 0.3]		Data with leakage in [0.3 0.9]	
	Case 1	Case 2	Case 1	Case 2	Case 1	Case 2
FVR	0.0129	0.0216	0.0026	0.0193	0.0185	0.0305
CS-FVR	0.0077	0.0254	0.0046	0.0257	0.0095	0.0240
WF-CS-FVR	0.0064	0.0245	0.0019	0.0246	0.0088	0.0232

test dataset in Case 1 is closer to the training dataset than in case 2, thus, supervised-learning methods give better results.

CS-FVR and WF-CS-FVR give better results than FVR on the data points in high-level degradation states, and worse results for data points in low-level degradation states: this shows that by sacrificing accuracy on the data points in low-level degradation states, the accuracy on the data points with high-level degradation states can be improved.

Compared to the results given by CS-FVR, by integrating the weighted-feature strategy in CS-FVR, the results are further improved on the data points with both low-level and high-level degradation states. Thus, in the case study, the weighted-feature strategy works well.

4. Conclusion

In practice, considering the imbalance of data and the different needs of the practitioners on the accuracy of the estimates of the degradation states, the traditional data-driven models for degradation assessment aiming at minimizing the average estimation error on the whole training dataset may give unsatisfactory results. In this paper, cost-sensitive and weighted-feature strategies are integrated in classical data-driven models (specifically, FVR in this work) to improve the results on imbalanced datasets. Differently from the approach of setting discrete cost values in the literature, continuous costs are assigned to the different data points and an efficient method is proposed for calculating the continuous costs with more focus on the peak values and high-level degradation states. In order to further improve the assessment accuracy, a weighted-feature strategy is also integrated to differentiate the contributions of different features. The application of the proposed approach on real data of leakage from the seals of RCPs in NPPs has been considered. Details for the parameters tuning are provided. The results show that the cost-sensitive strategy can improve the accuracy on the high-level degradation states and the weighted-feature strategy can further improve the accuracy both on high-level and low-level degradation states.

The proposed weighted-feature and cost-sensitive frameworks are not only suitable for FVR, but also applicable with other regression models, e.g. support vector machine, artificial neural network, Bayesian methods, etc. And the regression models can also be used for degradation assessment, degradation prediction and failure prediction, dependent on the input-output relation of the data.

References

[1] Alaswad S, Xiang Y. A review on condition-based maintenance optimization models for stochastically deteriorating system. Reliab Eng Syst Saf 2017;157:54–63. doi:10.1016/j.ress.2016.08.009.

[2] Alcala-Fdez J, Sanchez L, Garcia S, Del Jesus MJ, Ventura S, Garrell JM, Otero J, Romero C, Bacardit J, Rivas VM, Fernandez JC. KEEL: a software tool to assess evolutionary algorithms for data mining problems. Soft Comput 2009;13(3):307–18. doi:10.1007/s00500-008-0323-y.

[3] Amutha AL, Kavitha S. Features based classification of images using weighted feature support vector machines. Int J Comput Appl 2011;26(10):23–9. doi:10.5120/3141-4335.

[4] Baudat G, Anouar F. Feature vector selection and projection using kernels. Neurocomputing 2003;55(1):21–38. doi:10.1016/S0925-2312(03)00429-6.

[5] Borji A, Ahmadabadi MN, Araabi BN. Cost-sensitive learning of top-down modulation for attentional control. Mach Vis Appl 2011;22(1):61–76. doi:10.1007/s00138-009-0192-0.

[6] Caesarendra W, Widodo A, Yang BS. Application of relevance vector machine and logistic regression for machine degradation assessment. Mech Syst Signal Process 2010;24(4):1161–71. doi:10.1016/j.ymssp.2009.10.011.

[7] Cherkassky V, Ma Y. Practical selection of SVM parameters and noise estimation for SVM regression. Neural Netw 2004;17(1):113–26. doi:10.1016/S0893-6080(03)00169-2.

[8] Do P, Voisin A, Levrat E, Iung B. A proactive condition-based maintenance strategy with both perfect and imperfect maintenance actions. Reliab Eng Syst Saf 2015;133:22–32. doi:10.1016/j.ress.2014.08.011.

[9] Elkan C. The foundations of cost-sensitive learning. In: Proceedings of the international joint conference on artificial intelligence. vol. 17, Issue 1. LAWRENCE ERLBAUM ASSOCIATES LTD; 2001. p. 973–978.

[10] Gómez JA, Álvarez S, Soriano MA. Development of a soil degradation assessment tool for organic olive groves in southern Spain. Catena 2009;79(1):9–17. doi:10.1016/j.catena.2009.05.002.

[11] Jiang L, Li C, Wang S. Cost-sensitive Bayesian network classifiers. Pattern Recognit Lett 2014;45:211–16. doi:10.1016/j.patrec.2014.04.017.

[12] Jin G, Matthews DE, Zhou Z. A Bayesian framework for on-line degradation assessment and residual life prediction of secondary batteries inspacecraft. Reliab Eng Syst Saf 2013;113:7–20. doi:10.1016/j.ress.2012.12.011.

[13] Jouin M, Gouriveau R, Hissel D, Péra MC, Zerhouni N. Joint Particle Filters Prognostics for Proton Exchange Membrane Fuel Cell Power Prediction at Constant Current Solicitation. IEEE Trans Reliab 2016;65(1):336–49.

[14] Kukar M, Kononenko I. Cost-sensitive learning with neural networks. ECAI 1998:445–9.

[15] Lee W, Fan W, Miller M, Stolfo SJ, Zadok E. Toward cost-sensitive modeling for intrusion detection and response. J Comput Secur 2002;10(1–2):5–22. doi:10.3233/JCS-2002-101-202.

[16] Lei Y, Jia F, Lin J, Xing S, Ding SX. An intelligent fault diagnosis method using unsupervised feature learning towards mechanical big data. IEEE Trans Ind Electron 2016;63(5):3137–47.

[17] Liu M, Miao L, Zhang D. Two-stage cost-sensitive learning for software defect prediction. IEEE Trans Reliab 2014;63(2):676–86. doi:10.1109/TR.2014.2316951.

[18] Liu J, Vitelli V, Zio E, Seraoui R. A novel dynamic-weighted probabilistic support vector regression-based ensemble for prognostics of time series data. IEEE Trans Reliab 2015;64(4):1203–13. doi:10.1109/TR.2015.2427156.

[19] Liu J, Seraoui R, Vitelli V, Zio E. Nuclear power plant components condition monitoring by probabilistic support vector machine. Ann Nucl Energy 2013;56:23–33. doi:10.1016/j.anucene.2013.01.005.

[20] Liu J, Zio E. Feature vector regression with efficient hyperparameters tuning and geometric interpretation. Neurocomputing 2016;218:411–22. doi:10.1016/j.neucom.2016.08.093.

[21] Liu J, Zio E. A SVR-based ensemble approach for drifting data streams with recurring patterns. Appl Soft Comput 2016;47:553–64. doi:10.1016/j.asoc.2016.06.030.

[22] Mahi M, Baykan ÖK, Kodaz H. A new hybrid method based on particle swarm optimization, ant colony optimization and 3-opt algorithms for traveling salesman problem. Appl Soft Comput 2015;30:484–90. doi:10.1016/j.asoc.2015.01.068.

[23] Marseguerra M, Zio E, Podofillini L. Condition-based maintenance optimization by means of genetic algorithms and Monte Carlo simulation. Reliab Eng Syst Saf 2002;77(2):151–65. doi:10.1016/S0951-8320(02)00043-1.

[24] Moghaddass R, Zuo MJ. Multistate degradation and supervised estimation methods for a condition-monitored device. IIE Trans 2014;46(2):131–48. doi:10.1080/0740817X.2013.770188.

[25] Morshuis P. Assessment of dielectric degradation by ultrawide-band PD detection. IEEE Trans Dielectr Electr Insul 1995;2(5):744–60. doi:10.1109/94.469971.

[26] Oh H, Choi S, Kim K, Youn BD, Pecht M. An empirical model to describe performance degradation for warranty abuse detection in portable electronics. Reliab Eng Syst Saf 2015;142:92–9. doi:10.1016/j.ress.2015.04.019.

[27] Palshikar G. Simple algorithms for peak detection in time-series. In: Proceedings of the 1st international conference on advanced data analysis. Business Analytics and Intelligence; 2009. p. 1–13.

[28] Peng W, Li YF, Yang YJ, Huang HZ, Zuo MJ. Inverse Gaussian process models for degradation analysis: a Bayesian perspective. Reliab Eng Syst Saf 2014;130:175–89. doi:10.1016/j.ress.2014.06.005.

[29] Peng L, Zhang H, Zhang H, Yang B. A fast feature weighting algorithm of data gravitation classification. Inf Sci 2017;375:54–78. doi:10.1016/j.ins.2016.09.044.

[30] Pham HT, Yang BS, Nguyen TT. Machine performance degradation assessment and remaining useful life prediction using proportional hazard model and support vector machine. Mech Syst Signal Process 2012;32:320–30. doi:10.1016/j.ymssp.2012.02.015.

[31] Phan AV, Le Nguyen M, Bui LT. Feature weighting and SVM parameters optimization based on genetic algorithms for classification problems. Appl Intell 2016:1–15. doi:10.1007/s10489-016-0843-6.

[32] Rasmekomen N, Parlikad AK. Condition-based maintenance of multi-component systems with degradation state-rate interactions. Reliab Eng Syst Saf 2016;148:1–10. doi:10.1016/j.ress.2015.11.010.

[33] Razavi-Far R, Davilu H, Palade V, Lucas C. Model-based fault detection and isolation of a steam generator using neuro-fuzzy networks. Neurocomputing 2009;72(13):2939–51.

[34] Shen Z, He Z, Chen X, Sun C, Liu Z. A monotonic degradation assessment index of rolling bearings using fuzzy support vector data description and running time. Sensors 2012;12(8):10109–35. doi:10.3390/s120810109.

[35] Schölkopf B, Herbrich R, Smola AJ. A generalized representer theorem. In: Proceedings of the international conference on computational learning theory. Berlin Heidelberg: Springer; 2001. p. 416–426. DOI: 10.1007/3-540-44581-1_27.

[36] Si XS, Hu CH, Zio E, Li G. Modeling for prognostics and health management: methods and applications. Math Probl Eng 2015:2015. doi:10.1155/2015/613896.
[37] Vale C, Lurdes SM. Stochastic model for the geometrical rail track degradation process in the Portuguese railway Northern Line. Reliab Eng Syst Saf 2013;116:91–8. doi:10.1016/j.ress.2013.02.010.
[38] Xu J, Cao Y, Li H, Huang Y. Cost-sensitive learning of SVM for ranking. In: Proceedings of the European conference on machine learning. Berlin Heidelberg: Springer; 2006. p. 833–840. DOI: 10.1007/11871842_86.
[39] Yang F, Wang HZ, Mi H, Cai WW. Using random forest for reliable classification and cost-sensitive learning for medical diagnosis. BMC Bioinform. 2009;10(1):1. doi:10.1186/1471-2105-10-S1-S22.
[40] Zadrozny B, Langford J, Abe N. Cost-sensitive learning by cost-proportionate example weighting. In: Proceedings of the third IEEE international conference on data mining. ICDM 2003; 2003. p. 435–442. DOI: 10.1109/ICDM.2003.1250950.
[41] Zhang S, Qin Z, Ling CX, Sheng S. "Missing is useful": missing values in cost-sensitive decision trees. IEEE Trans Knowl Data Eng 2005;17(12):1689–93. doi:10.1109/TKDE.2005.188.
[42] Zhang Y, Zhou ZH. Cost-sensitive face recognition. IEEE Trans Pattern Anal Mach Intell 2010;32(10):1758–69. doi:10.1109/TPAMI.2009.195.
[43] Zhao W, Tao T, Zio E. System reliability prediction by support vector regression with analytic selection and genetic algorithm parameters selection. Appl Soft Comput 2015;30:792–802. doi:10.1016/j.asoc.2015.02.026.
[44] Zhao W, Tao T, Zio E, Wang W. A novel hybrid method of parameters tuning in support vector regression for reliability prediction: particle swarm optimization combined with analytical selection. IEEE Trans. Reliab. 2016;65:1393–405. doi:10.1109/TR.2016.2515581.
[45] Zhu X, Zhang Y, Zhu Y. Bearing performance degradation assessment based on the rough support vector data description. Mech Syst Signal Process 2013;34(1):203–17. doi:10.1016/j.ymssp.2012.08.008.
[46] Zio E. Prognostics and health management of industrial equipment. Diagn Progn Eng Syst: Methods Tech 2012:333–56.