An ensemble classifier for vibration-based quality monitoring

Vahid Yaghoubi[1,2], Liangliang Cheng[1,2], Wim Van Paepegem[1], Mathias Kersemans[1]

[1]Mechanics of Materials and Structures (MMS), Ghent University, Technologiepark 46, B-9052 Zwijnaarde, Belgium.

[2]SIM M3 program, Technologiepark 48 , B-9052 Zwijnaarde, Belgium.

Abstract

Vibration-based quality monitoring of manufactured components often employs pattern recognition methods. Albeit developing several classification methods, they usually provide high accuracy for specific types of datasets, but not for general cases. In this paper, this issue has been addressed by developing a novel ensemble classifier based on the Dempster-Shafer theory of evidence. In the proposed procedure, prior to DST combination, three steps should be taken: (i) selection of proper classifiers by maximizing the joint mutual information between predicted and target outputs, (ii) optimal redistribution of the classifiers' outputs by considering the distance between the predicted and target outputs, (iii) utilizing five different weighting factors to enhance the fusion performance. The effectiveness of the proposed framework is validated by its application to 13 UCI and KEEL machine learning datasets. It is then applied to two vibration-based datasets to detect defected samples: one synthetic dataset generated from the finite element model of a dogbone cylinder, and one real experimental dataset generated by collecting broadband vibrational response of polycrystalline Nickel alloy first-stage turbine blades. The investigation is made through statistical analysis in presence of different noise levels. Comparing the results with those of five state-of-the-art fusion techniques reveals the good performance of the proposed ensemble method.

Keywords: Ensemble classifier; Quality monitoring; Classifier selection, Classifier fusion; Dempster-Shafer theory of evidence; Joint mutual information

1 Introduction

Quality monitoring is a crucial step in any manufacturing process. In this regard, several methods have been developed based on X-ray CT and µCT [1–3], ultrasonic [4], thermography [5,6], and also vibration-based methods [7,8]. Extensive reviews of different methods can be found in [9,10]. Nowadays, (statistical) pattern recognition algorithms have been employed increasingly in quality monitoring due to its faster and more reliable online procedure with less operator training time and cost [8,11–15].

In this regard, several algorithms have been developed to reach the highest possible classification accuracy, e.g. discriminant analyses, decision trees, neural networks, support vector machines, support vector data descriptors, etc. [16]. However, it is well-known that none of the classifiers can show high accuracies in all the applications and datasets due to the presence of outliers, noise, errors, nonlinearities, and data redundancy [17]. An ensemble of classifiers is a remedy for this problem that could be used to enhance the strengths of each classifier and compensate for their weaknesses [18–24]. To this end, two important questions should be investigated: i) how to select classifiers to maintain the information and have diversity, and ii) how to perform the fusion.

To deal with the first issue, several methods have been proposed in the literature, e.g. different classification methods, different types of features, different training samples, etc. [25]. Another approach is to generate a pool of classifiers and select some of them that their combination gives the best classification outcome. To this end, Several classifier selection/ranking criteria have been proposed in the literature [26–28]. In these approaches, classifiers are selected based on a trade-off between the accuracy and diversity of the classifiers in the ensemble. Unlike the accuracy, there is no consensus about a metric to measure the diversity. Therefore, several metrics have been proposed [29]. Another category of the methods used for classifier selection is based on information theory [30,31]. Although these approaches are very common for feature selection, they have not been widely used for classifier selection. In this paper, we have adopted an information-based approach based on joint mutual information for classifier selection.

Several methods have been developed to tackle the second issue. From basic elementary operations like sum, average, maximum, and minimum of the outputs [20] to more advanced forms like majority voting [32], multilayered perceptrons [33], Bayes combination [34], fuzzy integrals [35], and Dempster-Shafer theory of evidence (DST) [36]. It is shown in [17,25,37] that the DST method has advantages over other combination methods. For instance, in [25,37] it is shown that the Dempster-Shafer method could improve the classification accuracy to be higher than that of the best individual model provided that the individual

models are independent. On the other hand, in presence of conflicting evidences, the DST-based fusion could lead to counter-intuitive results. To deal with this issue, several combination rules have been developed such as Yager's rule [38], Dubois–Prade (DP) rule [39], and proportional conflict redistribution (PCR) rules [40]. However, most of these techniques increase the computational complexity of the problem such that it may become untractable for high-dimensional problems. Therefore, these methods are less attractive among researchers. The alternative is to apply some preprocessing techniques on the evidences to redistribute them by some means. This could reduce their conflict which, in turn, could improve the combination outcome. This is the focus of the current paper.

Having the two aforementioned questions in mind, most of the works focused only on one of them separately without considering the underlying relation between them. Therefore, the major goal of the current paper is to develop an information-based framework for generating an ensemble classifier to be used for quality monitoring. The framework consists of two main aspects: Classifier selection and classifier combination. The contribution of this study is thus twofold:

(i) A classifier ranking/selection procedure based on Joint mutual information

(ii) An improved DST method by optimal redistribution of the classifiers' outputs

Besides, the effect of five weighting factors, stemmed from the confusion matrix, on improving the performance of the fusion method is exploited.

The paper is organized as follows. In Section 3 some required background about the Dempster-Shafer theory is explained. In Section 4 different steps of the proposed algorithm are elaborated. In Section 5 the proposed framework is first applied to 13 UCI and KEEL benchmark datasets and then to two datasets collected from the vibration behavior of (i) finite element model of a dogbone cylinder (synthetic dataset), and (ii) first-stage turbine blades with complex geometry and various damage features (real experimental dataset). In Section 6 concluding remarks are presented.

2 Related work

Ensemble learning is being more common among researchers to improve the performance of their learning tasks. Among different ensemble techniques, DST-based methods could improve the performance more than the others provided that proper classifiers with small conflicts are selected. To achieve this, one could redistribute the classifiers' outputs and/or select proper classifiers.

In this regard, Deng et. al. employed information from the confusion matrix to improve the basic belief assignment (BBA) [41]. In [42], Qian et. al. proposed to employ Shanon's information entropy together with Fuzzy preference relations (FPR) to reduce the conflict between the evidences. Xiao proposed to adjust the distance-based support degree of the evidences (SD) by utilizing the belief entropy and FPR [43]. This has been then used as the weight for evidences before applying Dempster's rule of combination. In [44], the evidences were weighted by using a criterion based on the similarity between the BBAs and the belief entropy. Dempster's rule of combination has been then applied to them. In [45], a novel method has been developed to evaluate the BBAs based on the *k*-nearest neighbor algorithm. The authors recently proposed a preprocessing technique based on Jousselme's distance and Belief Jensen–Shannon divergence [24]. Their combination with Deng entropy provides a new criterion to evaluate the credibility of the classifiers' outputs. The credibilities are then used as a weight for the classifiers prior to combining them. Wang et al. proposed to employ two weights for the evidences before combination [46]: One based on the credibility of the evidences and another one based on the support degree of the evidences with respect to the element with the largest mass values. In [47], by evaluating the reliability within a classifier and combine it with the relative reliability of classifiers, contextual reliability has been developed. This is applied to the classifiers' outputs prior to the DST combination. Liu et. al. has proposed a credal belief redistribution as a new technique to recast the classifiers' outputs before their combination [48]. In [49], an optimally weighted combination of classifiers based on evidential reasoning has been proposed.

As mentioned, another alternative is to select proper classifiers [27–29,50]. In [50], a diversity criterion based on Q-statistics has been proposed. This in conjunction with accuracy provided a foundation for a selection technique based on accuracy and diversity (SAD). In [29], pairwise accuracy and diversity of the classifiers have been used for their rankings. To evaluate the diversity, a combination of several criteria has been used. Graph theory has been recently employed to select proper classifiers from a pool [27,28]. Also, attempts have been made to choose proper classifiers by using information-related criteria [30,31], although this is not well established yet.

Since these two approaches are complementary to each other, the best solution would be their combination. However, there is limited literature that addresses both approaches [51]. In this paper, an information-based framework has been proposed to select proper classifiers and then combine them to achieve the highest possible accuracy.

3 Dempster-Shafer theory of evidence

Dempster-Shafer theory (DST) of evidence [36], is an important method for uncertain reasoning. It could be used to combine the uncertain information coming from different sources. It is defined as follows.

Let $\theta_1, \theta_2, \dots, \theta_K$ be a finite number of possible hypotheses describing a phenomenon. A set with all these hypotheses is called *Frame of discernment*, i.e. $\Theta = \{\theta_1, \theta_2, \dots, \theta_K\}$. Its powerset denoted by 2^Θ is a set of all its subsets including the null set φ and itself Θ. Each element of the powerset is called a proposition.

Basic belief assignment (BBA), $m(\cdot)$, is a function that maps the propositions to the range [0, 1] with the following conditions,

$$\begin{cases} m(\varphi) = 0 \\ \sum_{\mathcal{A} \in 2^\Theta} m(\mathcal{A}) = 1 \end{cases} \tag{1}$$

It should be mentioned that in belief theory a value can be assigned to a composite hypothesis $\mathcal{A}$, i.e. $\mathcal{A} = \{\theta_1, \theta_2\}$ without any overcommitment to either. This means $m(\mathcal{A}) + m(\bar{\mathcal{A}}) \leq 1$ in which $\bar{\mathcal{A}}$ is the complement of $\mathcal{A}$. Further, $1 - \big(m(\mathcal{A}) + m(\bar{\mathcal{A}})\big)$ is called "ignorance" associated with $\mathcal{A}$.

Combination rule provides a methodology to combine BBAs m_1 and m_2 in the frame of discernment Θ. It is defined as

$$m(\mathcal{A}) = (m_1 \oplus m_2)(\mathcal{A}) = \begin{cases} \dfrac{m_{12}(\mathcal{A})}{1 - m_{12}(\varphi)} & \mathcal{A} \subset \Theta, \mathcal{A} \neq \varphi \\ 0 & \mathcal{A} = \varphi \end{cases} \tag{2}$$

in which $\oplus$ is the orthogonal sum,

$$m_{12}(\mathcal{A}) = \sum_{\mathcal{C}_1 \cap \mathcal{C}_2 = \mathcal{A}} m_1(\mathcal{C}_1) m_2(\mathcal{C}_2) \tag{3}$$

indicates the conjunctive consensus on $\mathcal{A}$ among the sources $\mathcal{C}_1$ and $\mathcal{C}_2$. Besides, the denominator is the normalization factor in which,

$$m_{12}(\varphi) = \textstyle\sum_{\mathcal{C}_1 \cap \mathcal{C}_2 = \varphi} m_1(\mathcal{C}_1) m_2(\mathcal{C}_2), \tag{4}$$

could be used as a measure to estimate the conflict between the BBAs. This can be extended when we have several BBAs as,

$$m(\mathcal{A}) = \left(\left((m_1 \oplus m_2) \oplus m_3\right) \ldots \oplus m_n\right)(\mathcal{A}) \quad (5)$$

The combination rules (2) and (5) provide the required foundation for classifier fusion as explained in Section4.3.

4 Ensemble methodology

Figure 1 shows the building block of the proposed classifier ensemble method. It consists of three main stages:

I) Classifier generation: a pool of classifiers is the outcome of this step. This is explained in Section 4.1.

II) Classifier selection: in this step, proper classifiers are selected to be used in the ensemble. This is elaborated in Section 4.2.

III) Classifier fusion: in this step, the selected classifiers are combined to perform a fusion decision. This is extensively discussed in Section 4.3.

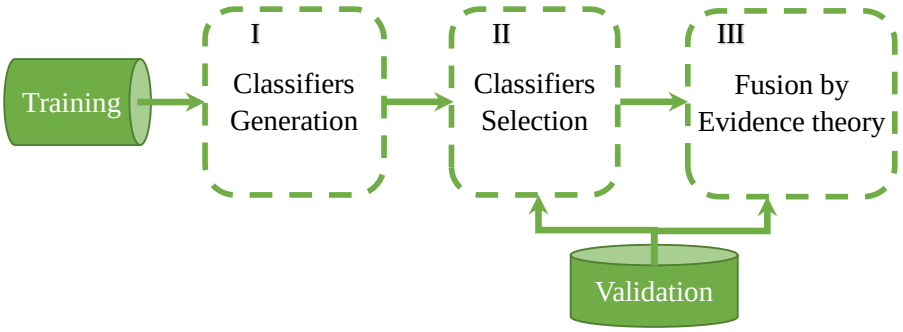

Figure 1. Flowchart of the ensemble-based quality monitoring procedure.

4.1 Classifiers generation

Several classifiers are generated to create a pool of classifiers. In this regard, eight classification methods are utilized. They are Decision tree (DT), Naïve Bayes (NB), Linear Discriminant Analysis (LDA), *k*-nearest neighbors (kNN), Support vector machines (SVM), and neural network (NN). A description of these methods can be found in [16]. Support vector data description (SVD) [52] is also employed. All the classifiers are implemented using the deep learning toolbox in Matlab 2019a. The parameters associated with each method have been estimated by using 10-fold cross-validation unless otherwise mentioned explicitly.

To evaluate the performance of the trained classifiers the so-called confusion matrix, see Figure 2, is implemented. Based on this matrix, several decisive measures have been developed that are presented in Table 1. The main measure here is accuracy (Acc) that indicates the portion of the samples that are correctly classified, Eq. (6). The other measures are precision (Pre) and recall (Rec) which are defined for the class k as Eq. (7), and Eq. (8) respectively.

$$Acc = \frac{\sum_{k=1}^{n_c} N_{kk}}{\sum_{i=1}^{n_c}\sum_{j=1}^{n_c} N_{ij}} \tag{6}$$

$$Pre_k = \frac{N_{kk}}{\sum_{i=1}^{n_c} N_{ik}} \tag{7}$$

$$Rec_k = \frac{N_{kk}}{\sum_{j=1}^{n_c} N_{kj}} \tag{8}$$

		True condition		
		c_1	$\cdots \quad c_k \quad \cdots$	c_{n_c}
Predicted condition	$\hat{c}_1$	N_{11}	$\cdots \quad N_{1k} \quad \cdots$	N_{1n_c}
	$\hat{c}_k$	N_{k1}	$\cdots \quad N_{kk} \quad \cdots$	N_{kn_c}
	$\hat{c}_{n_c}$	N_{n_c1}	$\cdots \quad N_{kn_c} \quad \cdots$	$N_{n_cn_c}$

Figure 2. Confusion matrix

4.2 Classifier selection:

To select proper classifiers for combination, a naïve approach is to search over all possible combinations between the classifiers in a pool to find the best ensemble [24]. Although this approach is very time-consuming, the existence of the best ensemble model could be guaranteed. This can also be achieved by evaluating and monitoring the joint mutual information between models' outputs and the true outputs [30].

To this end, let $\boldsymbol{X} = (x_1, x_2, \dots, x_N)$ be a discrete random variable. Its entropy $H(\boldsymbol{X})$ indicates the available uncertainty in the data which can be used to measure its information content. It is defined as

$$H(\boldsymbol{X}) = -\Sigma_{i=1}^{N} p(x_i)\log\big(p(x_i)\big) \tag{9}$$

in which $p(x_i)$ is the probability mass function. Mutual information (MI) is the amount of information shared between two random variables $\boldsymbol{X}$ and $\boldsymbol{\Omega} = (\omega_1, \omega_2, \dots, \omega_N)$. It is defined as,

$$MI(\boldsymbol{X}; \boldsymbol{\Omega}) = H(\boldsymbol{\Omega}) - H(\boldsymbol{\Omega}|\boldsymbol{X}) \tag{10}$$

Here, $H(\boldsymbol{\Omega}|\boldsymbol{X})$ is called conditional entropy and indicates the amount of information left in $\boldsymbol{\Omega}$ after introducing $\boldsymbol{X}$. It is defined as,

$$H(\boldsymbol{\Omega}|\boldsymbol{X}) = -\Sigma_{j=1}^{M}\Sigma_{i=1}^{N} p(x_i, \omega_j) \log\left(p(\omega_j|x_i)\right) \tag{11}$$

Mutual information could also be written as,

$$MI(\boldsymbol{X}; \boldsymbol{\Omega}) = \Sigma_{j=1}^{M}\Sigma_{i=1}^{N} p(x_i, \omega_j) \log\left(\frac{p(x_i, \omega_j)}{p(x_i)p(\omega_j)}\right) \tag{12}$$

Now, let $\mathbf{Z} = (z_1, z_2, \dots, z_N)$, be another random variable, the Joint mutual information (JMI) and Conditional mutual information (CMI) are defined as:

$$JMI(\mathbf{X}, \mathbf{Z}; \boldsymbol{\Omega}) = CMI(\mathbf{X}; \boldsymbol{\Omega}|\mathbf{Z}) + MI(\boldsymbol{\Omega}; \mathbf{Z}) \tag{13}$$

$$CMI(\mathbf{X}; \boldsymbol{\Omega}|\mathbf{Z}) = \Sigma_{k=1}^{K}\Sigma_{j=1}^{M}\Sigma_{i=1}^{N} p(x_i, \omega_j, z_k) \log\left(\frac{p(x_i, \omega_j, z_k)p(z_k)}{p(x_i, z_k)p(\omega_j, z_k)}\right) \tag{14}$$

let $\boldsymbol{Y} \in \mathbb{R}^{n_s \times n_c}$ be the true label in which, n_s, and n_c are the number of samples, and classes. Further, let $\widehat{\boldsymbol{Y}}_i, i = 1,2, \dots, N_c$ be the output predicted by the i^{th} classifier. Then $MI(\widehat{\boldsymbol{Y}}_i; \boldsymbol{Y})$ indicates the overlap information between the classifier's output and the true output $\boldsymbol{Y}$. This thus gives a bound on the accuracy and it is referred to as *relevancy* of the outputs to the target [30]. $JMI(\widehat{\boldsymbol{Y}}_i, \widehat{\boldsymbol{Y}}_j; \boldsymbol{Y}), i \neq j$, gives the overlap information between the true output $\boldsymbol{Y}$ and the union of $\widehat{\boldsymbol{Y}}_i$ and $\widehat{\boldsymbol{Y}}_j$. Eq. (13) indicates that it is calculated as the combination of overlap information between $\widehat{\boldsymbol{Y}}_i$ and $\boldsymbol{Y}$, i.e. $MI(\widehat{\boldsymbol{Y}}_i; \boldsymbol{Y})$, and the new information about $\boldsymbol{Y}$ that is brought by $\widehat{\boldsymbol{Y}}_j$, i.e. $CMI(\widehat{\boldsymbol{Y}}_j; \boldsymbol{Y}|\widehat{\boldsymbol{Y}}_i)$.

Now, if $\widehat{\mathbb{Y}} = \{\widehat{\boldsymbol{Y}}_1, \widehat{\boldsymbol{Y}}_2, \dots, \widehat{\boldsymbol{Y}}_{N_c}\}$, $\boldsymbol{Y}$ be a set of classifiers' output and the true labels respectively, the objective would be to maximize the $JMI(\widehat{\mathbb{Y}}; \mathbf{Y})$. This can be expanded as [30],

$$\begin{aligned} JMI(\widehat{\mathbb{Y}}; \mathbf{Y}) &= JMI(\widehat{\boldsymbol{Y}}_1, \widehat{\boldsymbol{Y}}_2, \dots, \widehat{\boldsymbol{Y}}_{N_c}; \mathbf{Y}) \\ &= \sum_{i=1}^{N_c} MI(\widehat{\boldsymbol{Y}}_i; \boldsymbol{Y}) - \sum_{\substack{\widehat{\mathbb{Y}} \subseteq \widehat{\mathbb{Y}} \\ |\widehat{\mathbb{Y}}| = 2, \dots, N_c}} MI(\{\widehat{\mathbb{Y}}\}) + \sum_{\substack{\widehat{\mathbb{Y}} \subseteq \widehat{\mathbb{Y}} \\ |\widehat{\mathbb{Y}}| = 2, \dots, N_c}} CMI(\{\widehat{\mathbb{Y}}\}|\boldsymbol{Y}) \end{aligned} \tag{15}$$

In which, the first term is the summation of mutual information between the individual classifiers' output and the target label. The second term which is independent of the true label evaluates the mutual information between all possible subsets of the classifiers. It indicates the overlap information between the classifiers therefore, it can be referred to as *ensemble redundancy*. The subtractive form of this term indicates its destructive role in ensemble performance. The last term is the conditional mutual information between classifiers' output and the true label. This is referred to as *conditional redundancy*. In general, this equation suggests that to have a good ensemble three conditions should be satisfied [30]: (1) high correlation between the classifiers and the true labels, (2) low correlation between the classifiers, and (3) high class-conditional correlation between the classifiers.

Although Eq. (15) gives good insights into the classifier selection procedure, it is clear that its evaluation in presence of a large number of classifiers and/or classifiers' outputs is not feasible. This leads to the development of several approaches to approximate it in a faster fashion [53,54]. Here, an approximation based on joint mutual information and maximum of minimum is used to rank the classifiers. It has been proved and shown in [54] that it approximates Eq. (15) and in [53] it has been used for feature selection. It is summarized in Algorithm 1. This procedure starts with the two classifiers' outputs that give the maximum $JMI(\hat{Y}_i, \hat{Y}_j; Y)$ [54]. Then, the other classifiers are selected based on the maximum of the minimum joint mutual information. After ranking the classifiers, the classifiers will be added to the ensemble one by one. At each step, the performance of the ensemble will be assessed and the best one will be selected.

Algorithm 1. The proposed classifier ranking approach

Inputs: $\hat{\mathbb{Y}} = \{\hat{Y}_1,\ \hat{Y}_2, \dots, \hat{Y}_{N_c}\}$, $\hat{\boldsymbol{Y}}^r = \{\emptyset\}$, $\boldsymbol{Y}$

$$[\hat{Y}_{s1}, \hat{Y}_{s2}] = \underset{\substack{\hat{Y}_j \in \hat{\mathbb{Y}} \\ \hat{Y}_i \in \hat{\mathbb{Y}}}}{argmax} \left(JMI(\hat{Y}_i, \hat{Y}_j; Y)\right),\ \hat{\boldsymbol{Y}}^r \leftarrow \{\hat{Y}_{s1}, \hat{Y}_{s2}\},\ \hat{\mathbb{Y}} \leftarrow \hat{\mathbb{Y}} \backslash \{\hat{Y}_{s1}, \hat{Y}_{s2}\}$$

while $\hat{\mathbb{Y}} \neq \{\emptyset\}$ **do**

$$\hat{Y}_s = argmax_{\hat{Y}_i \in \hat{\boldsymbol{y}}} \left(min_{\hat{Y}_j \in \hat{\boldsymbol{Y}}^r} \left(JMI(\hat{Y}_i, \hat{Y}_j; Y)\right)\right),$$

$$\hat{\boldsymbol{Y}}^r \leftarrow \hat{\boldsymbol{Y}}^r \cup \hat{Y}_s,$$

$$\hat{\mathbb{Y}} \leftarrow \hat{\mathbb{Y}} \backslash \hat{Y}_s$$

Output: $\hat{\boldsymbol{Y}}^r = \{\hat{Y}_1^r,\ \hat{Y}_2^r, \dots, \hat{Y}_{N_c}^r\}$

After ranking the classifiers, the best number of classifiers in the ensemble is selected by monitoring the accuracy of the ensemble assessed on the validation dataset. The procedure starts with the classifier with the highest MI, then the other classifiers will be added to the ensemble if their presence improves the

classification accuracy. The algorithm of the procedure is shown in Figure 3. The ensemble classifiers $\hat{\boldsymbol{Y}}^{en}$ with the best validation accuracy is the output of the procedure.

The dashed line that connects the training dataset to the validation path indicates the possibility of incorporating the training dataset to compare the performance of the classifiers. This could be done when several classifiers give the same validation accuracy. The situation mostly occurs when few samples are available for the validation dataset. To illustrate the procedure of classifier ranking procedure, a toy example has been provided in Appendix I.

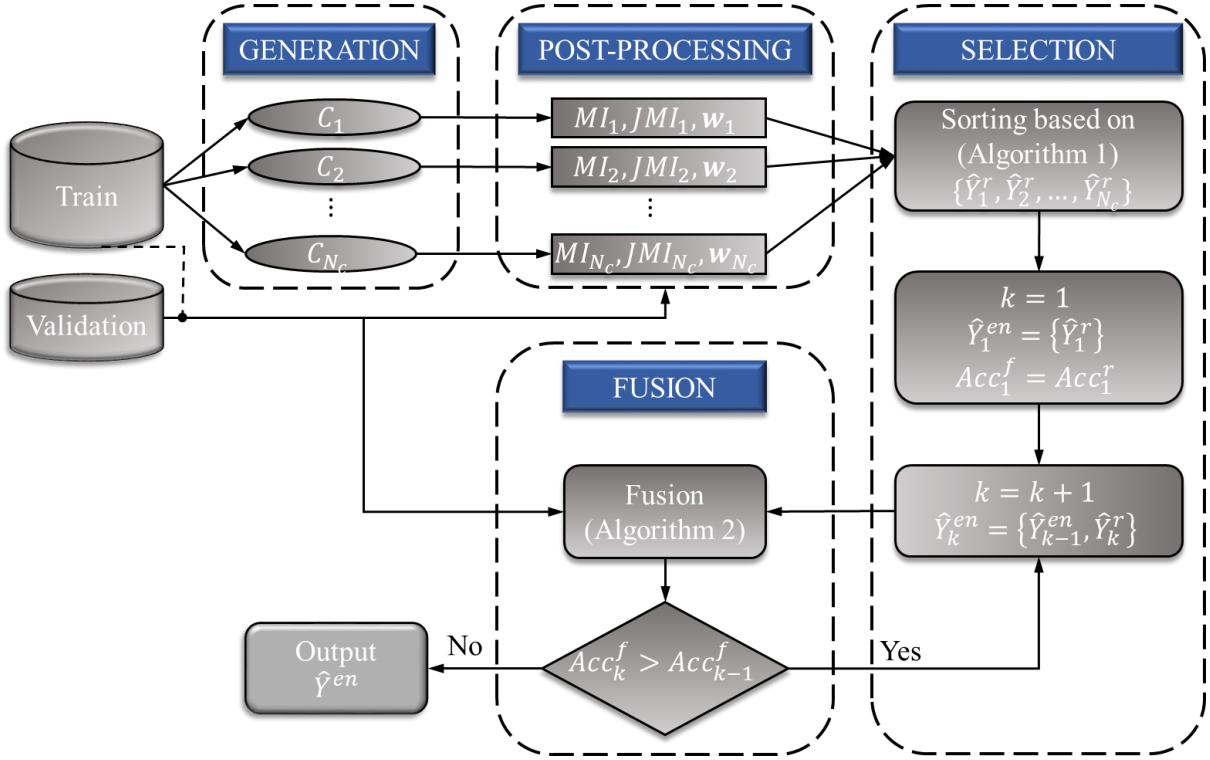

Figure 3. Flowchart of the proposed ensemble classifier. The ensemble classifiers $\hat{Y}^{en}$ with the best validation accuracy is the output of the procedure.

4.3 Classifier fusion by DST

In this section, the application of DST to classifier fusion is explained. In the literature, several variations of the DST have been proposed for classifier fusion [25,37,55,56]. In this regard, let $\Theta = \{\theta_1, \dots, \theta_k, \dots, \theta_{n_c}\}$ be the frame of discernment in which θ_k is the hypothesis that the sample $\boldsymbol{x}$ belongs to class k. For classifier $C_i, i = 1,2, \dots, N_c$, the associated ignorance and the belief in θ_k are indicated respectively by $m_i(\Theta)$ and $m_i(\theta_k)$. Further, let $\boldsymbol{w}_i \in \mathbb{R}^{1\times n_c}$ and $\boldsymbol{r}_i \in \mathbb{R}^{1\times n_c}$ be its weighting factor and reference vector. The BBAs will be combined based on the combination rule in Eqs. (2) and (5) to produce the new output vector $\boldsymbol{y}$ as follows,

$$\boldsymbol{y} = m(\theta_k) = m_1(\theta_k) \oplus \ldots \oplus m_{N_c}(\theta_k) \tag{16}$$

Different methods have been proposed to estimate the BBAs [25,37,55,56]. Here, a distance-based method has been implemented. In this method, the classifiers' outputs are redistributed based on their distance with a reference vector such that their combinations optimize the classification performance on the training and/or validation datasets. In this regard, let

$$\boldsymbol{D}_i = \phi(\boldsymbol{R}_i, \boldsymbol{w}_i \otimes \widehat{\boldsymbol{Y}}_{i,t}) = [\boldsymbol{d}_i^1, \ldots, \boldsymbol{d}_i^k, \ldots, \boldsymbol{d}_i^{n_c}] \in \mathbb{R}^{n_s \times n_c} \tag{17}$$

be a proximity measure between a reference matrix $\boldsymbol{R_i} = \boldsymbol{r}_i \otimes \mathbf{1} \in \mathbb{R}^{n_s \times n_c}$ and $\widehat{\boldsymbol{Y}}_{i,t}$. Here $\mathbf{1} \in \mathbb{R}^{n_s \times 1}$ is an all-one vector, $\otimes$ is the Kronecker product, and $(\cdot)_t$ stands for the training dataset. ϕ could be any function and/or norm that could represent this proximity. Here $\phi\left(\boldsymbol{R}_i, \boldsymbol{w}_i \otimes \widehat{\boldsymbol{Y}}_{i,t}\right) = \exp\left(-\left\|\boldsymbol{R}_i - \boldsymbol{w}_i \otimes \widehat{\boldsymbol{Y}}_{i,t}\right\|^2\right)$. Considering ε_i as the ignorance of C_i, BBAs are defined based on their distance to the reference vector as

$$m_i(\theta_k, \boldsymbol{R_i}, \varepsilon_i) = \frac{\boldsymbol{d}_i^k}{\sum_{k=1}^{K} \boldsymbol{d}_i^k + \varepsilon_i} \tag{18}$$

$$m_i(\Theta, \boldsymbol{R_i}, \varepsilon_i) = 1 - \sum_{k=1}^{n_c} m_i(\theta_k, \boldsymbol{R_i}, \varepsilon_i) = \frac{\varepsilon_i}{\sum_{k=1}^{K} \boldsymbol{d}_i^k + \varepsilon_i} \tag{19}$$

Eq. (18) gives the BBAs for the singleton classes after redistribution and Eq. (19) gives the BBA of the ignorance associated with the classifier. The next step is to obtain an optimal reference vector $\boldsymbol{R_i}$ and the ignorance ε_i that could improve the classification performance of the ensemble. This is achieved by minimizing the distance between the output of the ensemble model $\boldsymbol{y}(\boldsymbol{R}_i, \varepsilon; \widehat{\boldsymbol{Y}}_{i,t})$ defined in Eq.(16) and the true label $\boldsymbol{Y}_t$, i.e.

$$\boldsymbol{R}_i, \varepsilon = argmin(\|\boldsymbol{y}(\boldsymbol{R_i}, \boldsymbol{\varepsilon}) - \boldsymbol{Y}_t\|) \tag{20}$$

Weighting factor $\boldsymbol{w}$

The last step is to define a weighting factor $\boldsymbol{w}_i$. Five different weightings based on the confusion matrix (see Figure 2) are presented in Table 1. The $\boldsymbol{w}_0 = \mathbf{1} \in \mathbb{R}^{1 \times n_c}$, as a vector of ones, gives the unweighted version of the fusion algorithm. The other weightings $\boldsymbol{w}_i \in \mathbb{R}^{1 \times n_c}, i = 1,2, \ldots,5$ give a value in the range $[0, 1]$ to each class indicating the extent to which the class is reliable. In this table, $[\cdot]$ and $\odot$ are respectively used to show vectors and element-wise multiplications.

Algorithm 2. The proposed fusion method

Inputs: $\boldsymbol{T}$: $(\boldsymbol{X}_t, \boldsymbol{Y}_t)$ training dataset,

$\boldsymbol{V}$: $(\boldsymbol{X}_v, \boldsymbol{Y}_v)$ validation dataset,

$\widehat{\boldsymbol{Y}}_i$: output matrix predicted by the classifier C_i

$\boldsymbol{w}_i$: weighting factors based on Table 1

for $j = 1$ **to** 6 **do**

$D_{i,j} = \exp\left(-\left\|\boldsymbol{R}_{i,j} - w_{i,j} \otimes \widehat{\boldsymbol{Y}}_{i,t}\right\|^2\right)$

$m_i\left(\theta_k, \boldsymbol{R}_{i,j}, \varepsilon_{i,j}\right)$ by Eq. (18)

$m_i\left(\Theta, \boldsymbol{R}_{i,j}, \varepsilon_{i,j}\right)$ by Eq. (19)

Evaluate $\boldsymbol{y}_{j,t} = \boldsymbol{y}(\boldsymbol{R}_j, \boldsymbol{\varepsilon}_j; \widehat{\boldsymbol{Y}}_{i,t})$ using Eqs. (8), (5), and (2)

$\boldsymbol{R}_j, \boldsymbol{\varepsilon}_j = argmin\left(\left\|\boldsymbol{y}\left(\boldsymbol{R}_j, \boldsymbol{\varepsilon}_j; \widehat{\boldsymbol{Y}}_{i,t}\right) - \boldsymbol{Y}_t\right\|\right)$

$Acc_j = Acc(\boldsymbol{y}_{j,v}, \boldsymbol{Y}_v)$

i_{best} = index of maximum Acc_j

$\boldsymbol{R} = \boldsymbol{R}_j(i_{best}), \boldsymbol{\varepsilon} = \boldsymbol{\varepsilon}_j(i_{best}), \boldsymbol{w} = \boldsymbol{w}_j(i_{best})\ \widehat{\boldsymbol{Y}}^f = \boldsymbol{y}_{j,v}(i_{best})$

Output: $\boldsymbol{R}, \boldsymbol{\varepsilon}, \boldsymbol{w}, \widehat{\boldsymbol{Y}}^f$

Table 1. Weighting factors

Response	Weight	Formulation	Description	Reference
$\boldsymbol{F}_{w_0}^{\boldsymbol{p}}$	$\boldsymbol{w}_0$	$\mathbf{1}$	Unweighted form	---
$\boldsymbol{F}_{w_1}^{\boldsymbol{p}}$	$\boldsymbol{w}_1$	$Acc \otimes \mathbf{1}$	$Acc = \frac{\sum_{k=1}^{n_c} N_{kk}}{\sum_{i=1}^{n_c}\sum_{j=1}^{n_c} N_{ij}}$	
$\boldsymbol{F}_{w_2}^{\boldsymbol{p}}$	$\boldsymbol{w}_2$	$[Pre_1, \dots, Pre_k, \dots, Pre_{n_c}]$	$Pre_k = \frac{N_{kk}}{\sum_{i=1}^{n_c} N_{ik}}$	[57]
$\boldsymbol{F}_{w_3}^{\boldsymbol{p}}$	$\boldsymbol{w}_3$	$[Rec_1, \dots, Rec_k, \dots, Rec_{n_c}]$	$Rec_k = \frac{N_{kk}}{\sum_{j=1}^{n_c} N_{kj}}$	[58]
$\boldsymbol{F}_{w_4}^{\boldsymbol{p}}$	$\boldsymbol{w}_4$	$\boldsymbol{w}_2 \odot \boldsymbol{w}_3$	----	
$\boldsymbol{F}_{w_5}^{\boldsymbol{p}}$	$\boldsymbol{w}_5$	$\boldsymbol{w}_1 \odot \boldsymbol{w}_2$	----	[24]

The procedure of the fusion method is presented in Algorithm 2. In the grey area, the effect of different weightings on the performance of the fusion has been compared. The one with the best performance on the validation dataset is shown by i_{best} and its associated response will be referred to as the Best Ensemble Model (BEM). Since the effectiveness of the weighting factors depends on the application, this optional step is devised to select a weighting factor with the best effect on the fusion performance, but at the expense of higher computational time.

4.4 Prediction

To apply the method to a new sample, the following steps should be taken:

(i) Classify the sample by the selected individual classifiers C^{en}

(ii) Evaluate the BBAs (ms) by obtaining the proximity of the classifiers' responses and their associated references

(iii) Combine the BBAs by the Dempster's rule

Algorithm 3 presents the procedure to predict the health status of a new sample.

Algorithm 3. Class prediction of the new sample $\boldsymbol{x}_{new}$

Input: $\boldsymbol{x}_{new}$, $\boldsymbol{R}$, $\boldsymbol{\varepsilon}$, $\boldsymbol{w}$, trained classifiers C^{en}, (i_{best})

For all classifiers in C^{en}

Evaluate $\hat{\boldsymbol{y}}_i^s = C_i^{en}(\boldsymbol{x}_{new})$

for $j = 1$ **to** 6 **do**

$D_{i,j} = \exp\left(-\left\|\boldsymbol{R}_{i,j} - \boldsymbol{w}_{i,j} \otimes \hat{\boldsymbol{y}}_i^s\right\|^2\right)$

$m_i\left(\theta_k, \boldsymbol{R}_{i,j}, \varepsilon_{i,j}\right)$ by Eq. (18)

$m_i\left(\Theta, \boldsymbol{R}_{i,j}, \varepsilon_{i,j}\right)$ by Eq. (19)

Evaluate $\hat{\boldsymbol{y}}(\boldsymbol{R}_j, \boldsymbol{\varepsilon}_j)$ using Eqs. (8), (5), and (2)

If i_{best} available **do**

$\hat{\boldsymbol{y}} = \hat{\boldsymbol{y}}(\boldsymbol{R}_{i_{best}}, \boldsymbol{\varepsilon}_{i_{best}})$

Output: $\hat{\boldsymbol{y}}$

5 Application

In this section, the proposed classifier fusion method is first applied to thirteen well-known machine learning datasets. It is then applied to two datasets generated from vibrational responses: one synthetic dataset generated from the finite element model of a dogbone cylinder with varying geometrical and material properties, and one real experimental dataset generated by collecting broadband vibrational response of polycrystalline Nickel alloy first-stage turbine blades having a range of defect conditions.

5.1 Introduction

For each dataset, as mentioned in Section 4.1, 10 classifiers from the eight methods are selected to generate the pool of classifiers. Their hyper-parameters are set to their default in Matlab except for the following ones

(i) *k*NN with k = 5, 10, 15

(ii) SVD with "Gaussian" kernel and width parameters σ of 1 and 5. They are indicated respectively by SVD1 and SVD5. It should be mentioned that although the method has been originally developed as a one-class classifier [52], it can be extended to multi-class classifiers as presented in [46].

(iii) SVM with "Gaussian" kernels.

(iv) NN with one hidden layer, 10 neurons, and "tansig" activation function.

To train the classifiers, each dataset has been randomly divided into three parts: training, validation, and testing the classifiers. The classifiers have been trained by using the training dataset together with 10-fold cross-validation. The validation dataset has been used to monitor the accuracy of the ensembles. This has been done for two reasons: (i) to select a proper number of models in the ensemble, (ii) to select a proper weighting factor among different weightings i.e., $F_{w_0}^p$ to $F_{w_5}^p$, to generate BEM.

To highlight the performance of the fusion method in enhancing the classification, its outcome has been compared with 5 state-of-the-art DST-based fusion methods. For the sake of abbreviation, these methods are shown by F_1^L [44], F_2^L [59], F_3^L [43], F_4^L [46], and F_5^L [24].

5.2 Benchmark datasets from literature

In this section, the method has been applied to thirteen machine learning datasets collected from UCI [60] and KEEL [61]. Since the main interest of the authors is the development of a framework for quality monitoring (see next section), only datasets with two classes, i.e. $n_c = 2$, but with various imbalance ratios

(IR) have been chosen. The descriptions of the datasets are listed in Table 2. Here 50% of each dataset has been used for training, 25% for validation, and 25% for test the classifiers. The accuracy of the classifiers has been assessed when they are applied to the test dataset (see Table 3). For each dataset, the model with the highest accuracy is called Best Individual Model (BIM) and is shown in bold. Moreover, the trained classifiers have been ranked using the presented method. Their ranks are also shown in Table 3. The last row in the table shows the average classification accuracy of all datasets obtained by each method.

Table 2. Summary of the UCI and KEEL datasets. Here n_s is the number of samples, n_f is the number of features and IR is the imbalance ratio.

Label	**Name**	n_s	n_f	IR
1	Ovarian Cancer	216	63	1.27
2	Ionosphere	351	33	1.78
3	Wisconsin	683	9	1.86
4	Pima	768	8	1.87
5	Breast Cancer	699	9	1.9
6	Haberman	306	3	2.78
7	Vehicle 2	846	16	2.88
8	Glass	214	9	3.20
9	Yeast 3	1484	8	8.10
10	Ecoli 4	336	7	15.8
11	Abalone 9-18	731	8	16.4
12	Yeast 4	1484	8	28.1
13	Yeast 6	1484	8	41.4

Table 3. Accuracy (in percent) of different classification methods applied to UCI and KEEL datasets, assessed on the test set. BIMs are in bold.

		DT	NB	LDA	5NN	10NN	15NN	SVD1	SVD5	SVM	NN
1	Acc.	81.4	83.3	79.6	85.1	85.1	85.1	55.5	88.8	92.5	**94.4**
	Rank	10	2	3	5	6	7	9	8	4	1
2	Acc.	87.5	84.0	87.5	84.0	84.0	84.0	67.0	79.5	**93.1**	92.0
	Rank	3	10	6	7	9	4	8	2	1	6
3	Acc.	97.6	97.0	94.7	85.9	84.7	84.2	95.3	97.0	97.0	**99.4**
	Rank	6	7	8	5	3	9	4	10	2	1
4	Acc.	75.5	76.5	75.5	67.1	68.7	67.1	73.9	67.7	76.0	**78.6**
	Rank	5	9	10	6	8	2	4	7	3	1
5	Acc.	92.0	94.8	94.2	79.4	78.2	78.2	92.5	94.8	96.0	**96.5**
	Rank	7	2	9	5	3	4	10	6	8	1
6	Acc.	63.1	71.0	69.7	**75.0**	73.6	71.0	67.1	26.3	72.3	73.6
	Rank	7	5	1	3	10	8	4	9	6	2
7	Acc.	94.3	84.4	94.8	94.8	94.8	94.3	81.1	83.4	96.6	**97.6**
	Rank	8	4	9	7	6	5	2	10	3	1
8	Acc.	90.7	94.4	92.5	90.7	90.7	92.5	85.1	64.8	94.4	**98.1**
	Rank	10	9	2	3	7	5	4	6	8	1
9	Acc.	95.9	92.9	94.6	95.4	95.6	95.6	91.1	94.0	95.1	**96.7**
	Rank	8	3	6	9	10	2	4	5	7	1
10	Acc.	96.4	95.2	**97.6**	95.2	95.2	**97.6**	**97.6**	95.2	**97.6**	95.2
	Rank	9	10	1	2	4	3	6	8	5	7
11	Acc.	95.0	87.9	95.6	93.9	93.9	93.9	92.3	89.0	93.9	**96.1**
	Rank	6	7	3	10	8	9	4	5	2	1
12	Acc.	**97.5**	77.0	96.2	96.4	96.4	96.4	96.4	77.0	97.0	**97.5**
	Rank	6	10	4	3	1	5	2	8	7	9
13	Acc.	98.1	97.3	97.3	98.3	98.1	98.1	97.8	97.3	98.3	**98.6**
	Rank	6	10	7	3	1	8	9	4	5	2
Average		89.6	87.4	90.0	87.8	87.6	87.5	84.0	81.1	92.3	93.4

Table 4. Maximum Accuracy (in percent) on the test set of UCI and KEEL datasets obtained. Fusion method F_is are from the literature and $F^p_{w_i}$s are different variation of the proposed method. BEM stands for the Best Ensemble Model.

	Unweighted						**Weighted**					
	F^L_1	F^L_2	F^L_3	F^L_4	F^L_5	$F^p_{w_0}$	$F^p_{w_1}$	$F^p_{w_2}$	$F^p_{w_3}$	$F^p_{w_4}$	$F^p_{w_5}$	**BEM**
1	96.2	96.2	96.2	94.4	96.2	**98.2**	**98.2**	96.2	**98.2**	**98.2**	**98.2**	**98.2**
2	94.3	94.3	94.3	94.3	95.4	95.4	**96.6**	**96.6**	95.4	**96.6**	**96.6**	**96.6**
3	**99.4**	**99.4**	**99.4**	**99.4**	**99.4**	**99.4**	**99.4**	**99.4**	**99.4**	**99.4**	**99.4**	**99.4**
4	78.6	79.6	79.6	79.6	79.6	**81.3**	80.7	80.7	80.7	80.2	**81.3**	**81.3**
5	96.5	96.5	96.5	96.5	96.5	96.5	96.5	96.5	**97.1**	96.5	**97.1**	**97.1**
6	76.3	76.3	76.3	76.3	75.0	76.3	**77.6**	76.3	76.3	76.3	76.3	76.3
7	98.1	98.1	**98.5**	**98.5**	**98.5**	**98.5**	**98.5**	**98.5**	**98.5**	**98.5**	**98.5**	**98.5**
8	98.1	**100**	**100**	98.1	**100**	98.1	98.1	**100**	**100**	98.1	**100**	**100**
9	96.7	97.0	**97.3**	97.0	97.0	**97.3**	**97.3**	97.0	97.0	**97.3**	97.0	**97.3**
10	97.6	**100**	**100**	**100**	**100**	**100**	98.8	98.8	**100**	**100**	**100**	**100**
11	96.1	96.1	96.1	96.1	**96.7**	**96.7**	96.1	**96.7**	**96.7**	**96.7**	**96.7**	**96.7**
12	98.1	98.1	98.1	**98.4**	97.8	98.1	**98.4**	**98.4**	97.5	97.5	**98.4**	**98.4**
13	98.6	98.9	99.1	98.9	98.9	99.1	99.1	**99.5**	99.1	99.1	**99.5**	**99.5**
Sigwin 1	1	3	4	4	5	7	7	7	7	7	10	12
Sigwin 2	5	10	10	8	10	10	9	11	11	9	12	13
Avg(Acc)	94.2	94.6	94.7	94.4	94.7	95.0	95.0	95.0	95.1	95.0	95.3	95.3
Avg(rank)	8	5.6	4.1	5.4	5.1	2.8	3.4	3.0	3.1	3.3	1.4	1.1

The ranked classifiers have been added to the ensemble according to Figure 3, and the best ensemble based on its performance on the validation dataset has been selected. The accuracy of the best ensemble has been assessed on the test dataset and is shown in Table 4. The table is divided into two sections: unweighted, weighted. In the unweighted section, the fusion methods with weighting $\boldsymbol{w}_0 = \mathbf{1}$ have been employed. In the weighted section the weights $\boldsymbol{w}_1, \boldsymbol{w}_2, \dots, \boldsymbol{w}_5$ have been utilized to combine the classifiers. In each section, the maximum achieved accuracy is shown in bold. The ensembles that outperform the best individual model are underlined. Further, the four last rows of the table are devised to compare the performance of different fusion methods. “Sigwin1" indicates the number of times that each method gives the maximum classification accuracy that is, the number of bold accuracies in the column. “Sigwin2” shows the number of times that the fusion model outperforms the BIM that is, the number of underlined

accuracies in the column. “Avg(Acc)” shows the average accuracy of the fusion methods over all datasets. To obtain the last row, i.e. “Avg(rank)”, for each dataset, the fusion methods have been ranked based on their performance on improving the classification accuracy. The average of their ranking over all datasets provides the “Avg(rank)”.

Comparing the unweighted part of Table 4 with Table 3 reveals that for datasets three and five, all the methods gave the same performance and equal to their associated BIM. For the other 11 cases, at least one of the methods outperforms the BIM. Moreover, comparing the average accuracies reveals that, all the fusion methods improve the average classification accuracy, and among them $F_{w_0}^p$ improves the most.

By comparing different methods in the unweighted section of Table 4, one can see that the $F_{w_0}^p$ has the best performance in the sense of the maximum accuracy by having the “Sigwin 1” of 7. “Sigwin2” reveals that four of the fusion techniques could generate ensembles with classification the accuracy larger than that of the BIM in 10 datasets out of 13. Among them, $F_{w_0}^p$ gives the best average accuracy and average ranks.

Comparing different versions of the proposed methods, i.e. $F_{w_0}^p$ to $F_{w_5}^p$, reveals that $F_{w_5}^p$ gives the best classification performance in all four considered aspects, that is the maximum classification accuracy, outperforming the BIMs, average accuracy, and average rank. The next best ones are $F_{w_0}^p, F_{w_2}^p, F_{w_3}^p, F_{w_4}^p$, and $F_{w_1}^p$. The last column of the table presents the result for BEM. One can see that this approach gave the best classification performance by having an average ranking of 1.1 and average accuracy of 95.3%.

5.3 Application to the synthetic dataset of dogbone cylinders

In this section, the proposed method has been applied to a synthetic dataset generated from the finite element model of a dogbone cylinder. The samples are modeled as Nickel, single-crystal, cylindrical dogbones. The FE model of the cylinder is shown in Figure 4. The model contains eleven parameters: six geometrical parameters as shown in Figure 4, and five material parameters namely Density, Elastic modulus, Poisson’s ratio, Anisotropy ratio, and crystal orientation θ. Detail description of the model and its associated parameters can be found in [62].

To create a database, 3562 samples have been extracted from the parameter space and 44 resonance frequencies in the range of [3, 190] kHz have been estimated to be used as the features for the classification methods. To determine the quality of the samples two decisive factors have been chosen: $creep$ strain and crystal orientation θ. C$reep$ and θ have variations in the range [0, 9.9]% and [0, 37.9] respectively, and

samples have been called "damaged" provided that $creep > 0.5\%$ and $\theta > 10°$. This leads to 1857 healthy and 1750 damaged cylinders.

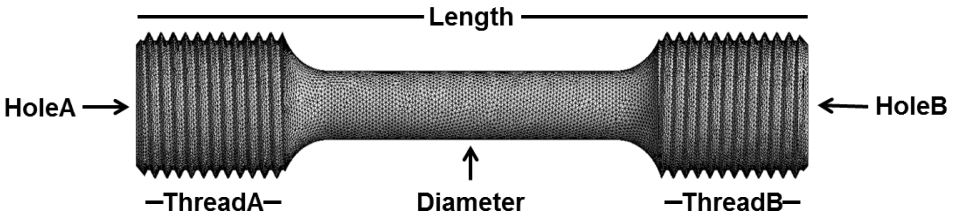

Figure 4. The finite element model of the dogbone cylinder. Its geometrical parameters are shown.

To commence the procedure 25% of the dataset has been used for training, 25% for validation, and 50% for test the classifiers. In the following, two different analyses have been carried out:

(i) Single model analysis: elaboration of one random train/validation/test dataset and the associated individual classifiers and their effects on the performance of the fusion models

(ii) Noise analysis: investigation on the effect of measurement noise on the fusion performance

To compare the performance of different methods, both the Acc (Eq. (6)) and the precision of each class (Eq. (7)) were investigated. From now on, they will be referred to as the True Positive Rate (TPR) for the healthy class and True Negative Rate (TNR) for the damaged class.

5.3.1 Single model analysis

The performance of the trained classifiers has been assessed on the training, validation, and test dataset in the sense of Accuracy, TNR, and TPR. The outcome is shown in Table 5. The ranking of the classifiers for adding into the ensemble is also presented in the table. It can be seen that BIM gave an accuracy of 99.10% on the test dataset. Now, based on Figure 3, the classifiers have been added to the ensemble and their performances on the validation and test datasets are respectively presented in Table 6 and Table 7. In Table 6, the column-wise comparison reveals the ensembles selected to be used for prediction for each method. They are shown in bold. The row-wise comparison between $F^p_{w_0}$ to $F^p_{w_5}$ indicates the ensembles selected to serve as the BEM. They are underlined. The accuracy of all ensemble models assessed on the test dataset is reported in Table 7. The bold and underlined models correspond to the same ones in Table 6.

It can be observed that based on the proposed classifier selection among the unweighted methods, in this example, F^L_1 to F^L_4 could not improve the performance of the BIM. Among different weighting factors, only $F^p_{w_1}$ could not improve the classification accuracy. The BEM could improve the classification accuracy to 99.3% on the test dataset.

Table 5. Accuracy of the classifiers assessed on the training, validation, and test datasets. The ranking of the classifiers for addition to the ensemble is also shown. Best Individual Model (BIM) is bolded.

		DT	**NB**	**LDA**	**11NN**	**13NN**	**15NN**	**SVD1**	**SVD5**	**SVM**	**NN**
Train	Acc	99.4	88.6	94.9	93.8	93.3	93.3	100	95.8	98.9	**98.2**
	TNR	99.0	84.0	89.6	95.0	95.0	94.8	100	92.7	97.8	**96.7**
	TPR	99.7	92.8	99.7	92.6	91.8	92.0	100	98.7	100	**99.5**
Valid.	Acc	94.1	85.0	95.0	94.8	94.7	94.4	77.4	94.2	97.3	**98.9**
	TNR	92.9	77.9	90.3	96.4	96.2	96.2	98.5	93.6	97.1	**98.8**
	TPR	95.2	91.5	99.3	93.3	93.3	92.8	57.9	94.8	97.4	**99.1**
Test	Acc	94.6	87.5	96.0	94.6	94.2	94.3	80.7	95.2	97.6	**99.1**
	TNR	93.0	81.8	92.1	97.6	97.4	97.4	99.1	94.1	97.4	**98.5**
	TPR	96.0	92.8	99.5	91.9	91.2	91.4	63.8	96.3	97.9	**99.5**
Rank		9	3	8	6	2	10	4	5	7	1

Table 6. Accuracy of the ensembles assessed on the validation dataset by increasing the number of Constituent Models (# CM). The ensembles selected for prediction are shown in bold. The weights selected to serve as the Best Ensemble Model (BEM) are underlined.

# CM	**Unweighted**						**Weighted**					
	F_1^L	F_2^L	F_3^L	F_4^L	F_5^L	$F_{w_0}^p$	$F_{w_1}^p$	$F_{w_2}^p$	$F_{w_3}^p$	$F_{w_4}^p$	$F_{w_5}^p$	**BEM**
1	**98.9**	**98.9**	**98.9**	**98.9**	98.9	98.9	98.9	98.9	98.9	98.9	98.9	98.9
2	95.7	97.8	98.9	98.9	**99.2**	**99.2**	98.9	98.9	**99.1**	**99.2**	**99.1**	**99.2**
3	93.1	98.5	98.6	98.2	98.9	98.8	98.9	**99.2**	98.9	98.7	98.8	**99.2**
4	76.2	96.6	96.6	98.4	98.9	98.7	**99.1**	98.9	98.7	97.9	98.8	99.1
5	76.1	97.5	97.5	98.2	98.2	98.7	98.3	97.8	98.8	98.7	98.6	98.8
6	73.9	97.7	97.5	97.6	98.4	98.8	98.3	96.8	98.8	98.8	94.3	98.8
7	74.0	97.7	97.5	97.8	98.4	98.7	97.1	99.1	93.8	97.1	99.1	99.1
8	74.0	97.8	98.2	98.2	98.4	96.9	98.7	97.5	97.9	98.7	98.3	98.7
9	73.7	97.9	97.9	97.9	98.3	97.9	98.0	98.2	98.9	98.9	96.1	98.9
10	71.2	97.9	98.0	97.7	98.2	93.8	97.7	98.9	98.3	98.9	98.6	98.9

Table 7. Accuracy of the ensembles assessed on the test dataset by increasing the number of Constituent Models (# CM). Accuracy of the ensembles selected for prediction is bold. The ensembles selected to serve as the Best Ensemble Model (BEM) are underlined. The last row is the Reference Model (Ref.) associated with each method.

# CM	**Unweighted**						**Weighted**					
	F_1^L	F_2^L	F_3^L	F_4^L	F_5^L	$F_{w_0}^p$	$F_{w_1}^p$	$F_{w_2}^p$	$F_{w_3}^p$	$F_{w_4}^p$	$F_{w_5}^p$	**BEM**
1	**99.1**	**99.1**	**99.1**	**99.1**	99.1	99.1	99.1	99.1	99.1	99.1	99.1	99.1
2	95.0	97.4	99.0	99.0	**99.2**	**99.3**	98.8	99.2	**99.3**	**99.3**	**99.3**	**99.3**
3	93.8	98.4	98.4	98.6	99.0	98.9	98.8	**99.2**	99.0	98.9	99.0	**99.2**
4	79.9	96.9	96.8	98.6	98.8	98.7	**99.1**	98.9	99.0	98.3	98.9	99.1
5	79.7	97.8	97.8	98.3	98.3	98.9	98.4	98.3	98.8	98.9	98.7	98.8
6	77.1	97.5	97.7	97.6	98.5	99.2	98.9	96.7	98.7	98.9	95.3	98.9
7	77.3	98.1	97.9	98.0	98.3	98.9	97.7	99.1	94.8	97.6	99.2	99.2
8	77.4	98.1	98.1	98.1	98.3	97.0	98.8	97.4	98.1	98.8	98.4	98.8
9	76.4	98.0	98.0	98.2	98.2	98.2	98.4	98.3	98.6	98.9	97.5	98.9
10	74.4	97.9	97.9	98.0	98.2	95.1	98.1	98.9	98.3	98.7	98.4	98.9
Ref	99.2	99.1	99.1	99.1	99.2	**99.4**	99.3	99.3	99.3	99.3	99.3	**99.4**

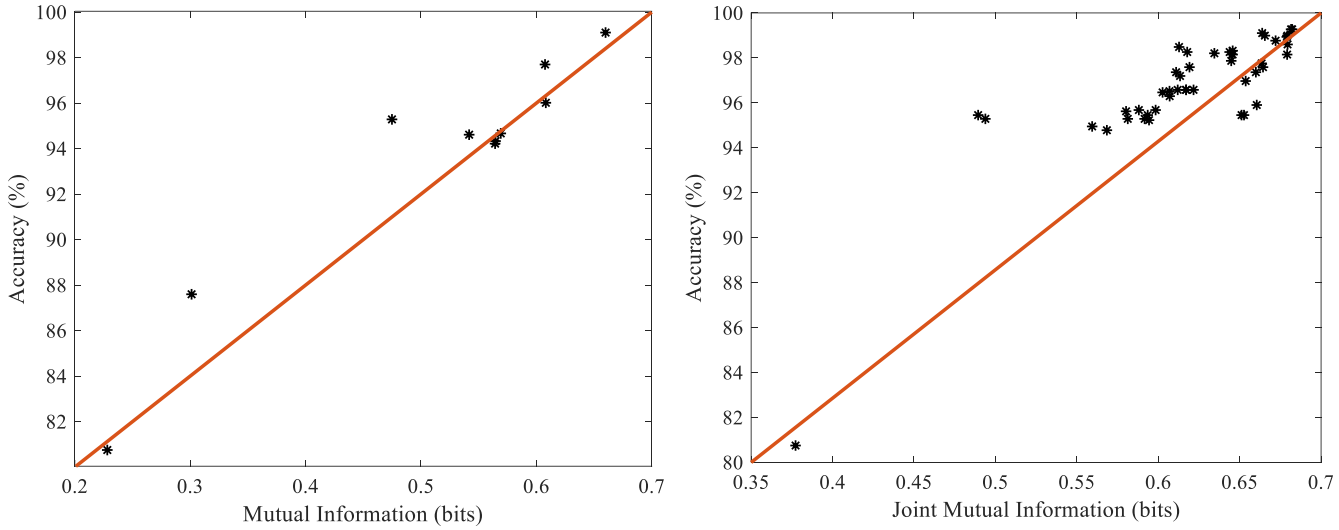

Figure 5. left) validation data mutual information versus test data accuracy of the individual classifiers, right) validation data joint mutual information versus test data accuracy of the two-classifier fusion model

The last row of Table 7 presents the maximum achievable classification accuracy by each method. They have been obtained by extensive search over all possible combinations between the individual classifiers [24]. From here on, such models will be called “Reference Models”. This row could reveal two important

points: (i) regardless of the classifier ranking/selection procedure, the proposed combination technique, i.e. $\boldsymbol{F}_{\boldsymbol{w_0}}^{\boldsymbol{p}}$, gives the highest classification accuracy among the considered ensemble methods. (ii) although the presented classifier selection technique could not achieve the optimal classification accuracy, it could improve the classification performance to a reasonably good suboptimal one which could save considerable time and memory.

Figure 5 demonstrates the efficacy of the joint mutual information for selecting proper classifiers. In the left plot, the test data accuracy of the individual classifiers has been shown as a function of the mutual information between their outputs and the true outputs of the validation dataset. Whereas, in the right plot, the test data accuracy of the fusion of two classifiers has been shown as a function of their joint mutual information with the true outputs. The results indicate the strong correlation between the joint mutual information and the test data accuracy. This implies the effectiveness of using this criterion for selecting proper classifiers.

For further comparison between different combination techniques, their performances in the sense of Accuracy, TPR, and TNR have been investigated through statistical analysis. In this regard, the whole procedure of data resampling, classifier training, classifier selection, and classifier combination has been repeated 25 times. The results are shown in Table 8. In the "reference" section, the best performance of the fusion methods has been listed whereas, in the "selected" section, the performance of the fusion method in conjunction with the proposed classifier selection has been presented. It implies that the proposed classifier selection-fusion framework can make suboptimal ensembles that could outperform the BIMs as well as other combination methods.

Table 8. Statistical analysis of the performance of Best Individual Model (BIM), classifier ensemble techniques, and Best Ensemble Model (BEM)), assessed on the test dataset. They were obtained through 25 repetitions of the whole procedure of data resampling, classifier training, selection, and combination.

Metric		**BIM**	$\boldsymbol{F}_1^{\boldsymbol{L}}$	$\boldsymbol{F}_2^{\boldsymbol{L}}$	$\boldsymbol{F}_3^{\boldsymbol{L}}$	$\boldsymbol{F}_4^{\boldsymbol{L}}$	$\boldsymbol{F}_5^{\boldsymbol{L}}$	**BEM**
Selected	Acc	98.9 ± 0.2	99.0 ± 0.2	98.9 ± 0.2	98.9 ± 0.2	99.0 ± 0.2	99.1 ± 0.1	99.3 ± 0.2
	TNR	98.8 ± 0.2	98.8 ± 0.3	98.8 ± 0.2	98.8 ± 0.2	98.8 ± 0.2	99.1 ± 0.2	99.3 ± 0.2
	TPR	99.1 ± 0.2	99.1 ± 0.2	99.1 ± 0.2	99.1 ± 0.2	99.2 ± 0.2	99.1 ± 0.2	99.3 ± 0.2
Reference	Acc	98.9 ± 0.2	99.0 ± 0.1	98.9 ± 0.2	99.0 ± 0.2	99.0 ± 0.2	99.1 ± 0.1	99.3 ± 0.1
	TNR	98.8 ± 0.2	98.8 ± 0.2	98.8 ± 0.2	98.8 ± 0.2	98.8 ± 0.2	99.1 ± 0.2	99.3 ± 0.2
	TPR	99.1 ± 0.2	99.1 ± 0.2	99.1 ± 0.2	99.2 ± 0.2	99.2 ± 0.2	99.1 ± 0.2	99.3 ± 0.2

5.3.2 Noise analysis

In this section, the robustness of the different ensemble techniques in presence of noise has been investigated. In this regard, the features have been first polluted with different levels of noise and then through 25 times iteration, uncertainty bounds for the accuracy, TNR, and TPR of the classifier ensembles have been estimated. At each iteration, the whole proposed procedure as shown in Figure 3, namely data resampling, classifier training, selection, and fusion, has been done independently. The results are presented in Table 9 only when the classifier selection method has been employed.

The analyses have been performed when the features have been polluted with four different noise levels, namely 1%, 2%, 5%, and 10% rms Noise-to-signal ratio (NSR). One can see that comparing to BIM, by using the proposed procedure, all the fusion methods could improve the classification performance. Among them, the proposed ensemble technique, i.e. BEM, gave the highest improvement. This could be expected since, in this approach, a proper weight could be chosen depending on the dataset in hand. Moreover, it can be seen that the proposed ensemble classifier is more effective in presence of higher noise levels. That is, in the noiseless case, i.e. Table 8, the BEM improved the classification accuracy from 98.9% for BIM to 99.3% whereas, in 10% rms NSR, it improved the accuracy from 98.0% to 98.6%.

Table 9. Statistical analysis of the performance of Best Individual Model (BIM), classifier ensemble techniques, and Best Ensemble Model (BEM), assessed on the test dataset. They were obtained through 25 repetitions of random data resampling, classifier training, selection, and combination in presence of different noise levels.

Noise levels	Metric	BIM	F_1^L	F_2^L	F_3^L	F_4^L	F_5^L	BEM
1	Acc	98.9 ± 0.3	98.9 ± 0.2	98.9 ± 0.2	99.0 ± 0.2	99.0 ± 0.2	99.1 ± 0.1	99.2 ± 0.2
	TNR	98.7 ± 0.5	99.0 ± 0.3	98.8 ± 0.4	98.9 ± 0.4	98.9 ± 0.4	99.1 ± 0.3	99.2 ± 0.2
	TPR	99.1 ± 0.2	98.9 ± 0.6	99.1 ± 0.2	99.1 ± 0.2	99.1 ± 0.2	99.0 ± 0.2	99.1 ± 0.2
2	Acc	98.8 ± 0.2	98.9 ± 0.1	98.8 ± 0.2	98.9 ± 0.1	98.9 ± 0.1	99.0 ± 0.1	99.1 ± 0.1
	TNR	98.6 ± 0.4	98.8 ± 0.3	98.6 ± 0.4	98.7 ± 0.3	98.7 ± 0.3	99.0 ± 0.3	99.1 ± 0.2
	TPR	99.0 ± 0.2	98.9 ± 0.4	99.0 ± 0.2	99.1 ± 0.2	99.2 ± 0.2	99.0 ± 0.2	99.1 ± 0.3
5	Acc	98.5 ± 0.5	98.6 ± 0.3	98.6 ± 0.3	98.7 ± 0.3	98.7 ± 0.3	98.8 ± 0.2	99.0 ± 0.2
	TNR	98.3 ± 0.7	98.6 ± 0.5	98.4 ± 0.6	98.5 ± 0.5	98.5 ± 0.5	98.9 ± 0.5	99.0 ± 0.4
	TPR	98.7 ± 0.4	98.6 ± 0.6	98.8 ± 0.4	98.9 ± 0.3	98.9 ± 0.3	98.8 ± 0.3	98.9 ± 0.3
10	Acc	98.0 ± 0.5	98.1 ± 0.4	98.1 ± 0.4	98.2 ± 0.4	98.2 ± 0.4	98.4 ± 0.3	98.6 ± 0.2
	TNR	97.7 ± 0.7	98.3 ± 0.5	97.6 ± 0.8	97.7 ± 0.8	97.7 ± 0.7	98.0 ± 0.5	98.2 ± 0.4
	TPR	98.3 ± 0.5	98.0 ± 0.8	98.5 ± 0.4	98.7 ± 0.4	98.7 ± 0.4	98.7 ± 0.4	98.8 ± 0.3

5.4 Application to First-stage turbine blade

In this section, the proposed classifier selection/fusion method has been applied to an experimental dataset. This dataset is a broadband vibrational response collected from equiax Polycrystalline Nickel alloy first-stage turbine blades with complex geometry and various damage features. The blade has a cooling channel in the middle that imposes geometrical complexities for its investigation, see its CAD models in Figure 6. The defected blades have damages from inter-granular attack (corrosion), airfoil cracking, microstructure changes due to over-temperature, thin walls due to casting, MRO operations, and/or service wear. Needless to mention that the health condition of the blades has been first evaluated by some means, e.g. X-ray, visual testing, penetrant testing, ultrasonics, operator experience, etc., and then, the classification methods applied to them.

The amplitude of their frequency response function (FRF) has been collected by using one actuator and two sensors in the range of [3, 38] kHz. 16 frequencies, $F_i\ i = 1,2,\dots,16$ and 16 quality factors, $Q_i\ i = 1,2,\dots,16$ have been extracted from the FRFs to be used as the features for the classifiers. The database has been created by measuring the FRF from 192 healthy and 33 defected blades, i.e. $IR = 5.82$.

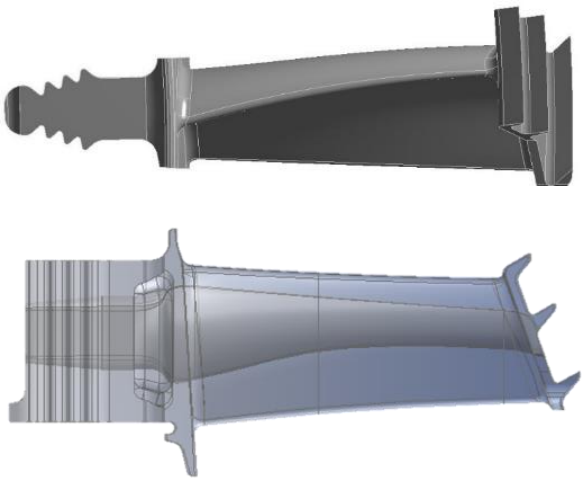

Figure 6. Two views of the CAD model of the Equiax Polycrystalline Nickel alloy first-stage turbine blade. The bottom plot shows a transparent view to illustrate the cooling channel.

Here, 50% of the dataset has been used for training, 25% for validating, and 25% for testing the classifiers. Table 10 presents the accuracy of different fusion techniques in comparison with the BIM when they are applied to the test dataset. As can be seen, the proposed classifier selection-fusion method could improve the classification performance from 96.4% for the BIM to 100% for the BEM.

To provide a deeper comparison between the fusion methods, statistical analysis has been devised by repeating the whole procedure 50 times. Results are presented in Table 11. The reference results indicate

that all the fusion methods could improve Acc, TNR, and TPR, among them the proposed fusion method improves the most. The "selected" section confirms that the proposed classifier selection method reaches an ensemble with suboptimal performance that could outperform the BIM.

Table 10. Effect of different ensemble techniques and the Best Ensemble Model (BEM) on the classification performance in comparison with Best Individual Model (BIM). They were assessed on the test dataset.

		BIM	F_1^L	F_2^L	F_3^L	F_4^L	F_5^L	$F_{w_0}^p$	$F_{w_1}^p$	$F_{w_2}^p$	$F_{w_3}^p$	$F_{w_4}^p$	$F_{w_5}^p$	**BEM**
Selected	Acc	96.4	96.4	96.4	96.4	96.4	96.4	98.2	98.2	96.4	**100**	98.2	**100**	**100**
	TNR	87.5	87.5	87.5	87.5	87.5	87.5	87.5	87.5	87.5	**100**	87.5	**100**	**100**
	TPR	97.9	97.9	97.9	97.9	97.9	97.9	100	100	97.9	**100**	100	**100**	**100**
Reference	Acc	96.4	96.4	96.4	96.4	98.2	98.2	**100**	98.2	98.2	**100**	98.2	**100**	**100**
	TNR	87.5	87.5	87.5	87.5	87.5	87.5	**100**	87.5	87.5	**100**	87.5	**100**	**100**
	TPR	97.9	97.9	97.9	97.9	100	100	**100**	100	100	**100**	100	**100**	**100**

Table 11. Statistical analysis of the performance of Best Individual Model (BIM), classifier ensemble techniques, and Best Ensemble Model (BEM), assessed on the test dataset. They were obtained through 50 repetitions of the whole procedure of data resampling, classifier training, selection, and combination.

Metric		**BIM**	F_1^L	F_2^L	F_3^L	F_4^L	F_5^L	**BEM**
Selected	Acc	96.5 ± 2.6	97.5 ± 1.5	97.6 ± 1.5	97.5 ± 1.5	97.6 ± 1.4	97.6 ± 1.5	98.4 ± 1.1
	TNR	85.0 ± 11.0	88.7 ± 9.2	89.5 ± 8.9	88.7 ± 9.5	89.7 ± 9.3	89.7 ± 9.0	93.0 ± 7.2
	TPR	98.4 ± 2.1	98.8 ± 1.4	98.9 ± 1.4	99.0 ± 1.4	99.0 ± 1.3	99.1 ± 1.3	99.3 ± 1.1
Reference	Acc	96.0 ± 2.6	98.3 ± 1.3	98.5 ± 1.3	98.3 ± 1.3	98.3 ± 1.2	98.3 ± 1.3	98.6 ± 1.2
	TNR	85.0 ± 11.0	92.7 ± 7.1	92.2 ± 7.9	92.0 ± 8.3	92.0 ± 8.3	92.0 ± 7.9	93.7 ± 7.2
	TPR	98.4 ± 2.1	99.3 ± 1.1	99.5 ± 1.0	99.4 ± 1.0	99.4 ± 1.0	99.3 ± 1.0	99.6 ± 0.8

To compare the robustness of the different approaches in presence of noise, the features have been polluted with four different synthetic noise levels, namely 1%, 2%, and 5% rms NSR. Here, 0% rms NSR means the features have been used as they had been measured, without adding any synthetic noise. However, since the data comes from real test data, it contains some levels of noise and/or error. To take that into account, a representative blade of the dataset has been measured 50 times, i.e. put a sample on the test-nest, perform the vibrational test, take the sample off the test-nest, and repeat. From this, the error of the typical measurement procedure has been estimated. This is referred to as the measurement error (ME).

The effect of ME on fusion performance has also been investigated. The results are presented in Table 12 only when the classifier selection method has been employed. As can be seen, all the fusion methods in conjunction with the selection method could outperform the BIM. Among them, the BEM could improve classification performance the most.

Table 12. Statistical analysis of the performance of Best Individual Model (BIM), classifier ensemble techniques, and Best Ensemble Model (BEM), assessed on the test dataset. They were obtained through 50 repetitions of random data resampling, classifier training, selection, and combination in presence of different noise levels and the Measurement error (ME).

Noise levels	**Metric**	**BIM**	F_1^L	F_2^L	F_3^L	F_4^L	F_5^L	**BEM**
1	Acc	95.4 ± 2.7	96.4 ± 1.9	97.2 ± 1.5	97.2 ± 1.6	97.1 ± 1.9	97.4 ± 1.6	98.0 ± 1.4
	TNR	84.2 ± 13.2	92.2 ± 7.8	90.2 ± 7.4	90.2 ± 7.4	90.0 ± 9.2	90.8 ± 8.9	92.8 ± 7.2
	TPR	97.8 ± 1.8	97.2 ± 2.0	98.6 ± 1.3	98.6 ± 1.3	98.5 ± 1.4	98.8 ± 1.4	99.0 ± 1.2
2	Acc	94.8 ± 3.1	95.7 ± 2.6	96.8 ± 2.1	96.9 ± 1.9	97.2 ± 1.8	97.4 ± 2.0	97.9 ± 1.9
	TNR	84.4 ± 12.1	92.2 ± 10.1	87.6 ± 9.8	88.2 ± 9.4	89.0 ± 9.0	89.8 ± 9.9	93.0 ± 8.6
	TPR	96.9 ± 3.2	96.4 ± 3.1	98.6 ± 1.5	98.7 ± 1.6	98.8 ± 1.6	98.9 ± 1.3	98.9 ± 1.5
5	Acc	94.9 ± 2.9	95.8 ± 2.9	96.6 ± 2.3	96.6 ± 2.1	96.7 ± 2.0	97.2 ± 2.1	97.6 ± 2.0
	TNR	82.0 ± 12.2	88.6 ± 11.2	85.2 ± 11.4	84.8 ± 10.9	84.8 ± 10.1	88.6 ± 10.5	89.8 ± 9.9
	TPR	97.5 ± 2.6	97.2 ± 2.4	98.8 ± 1.4	99.0 ± 1.2	99.2 ± 1.2	98.9 ± 1.2	99.2 ± 1.2
ME	Acc	95.6 ± 2.7	96.3 ± 2.4	97.5 ± 1.8	97.6 ± 1.9	97.4 ± 2.0	97.7 ± 1.6	98.1 ± 1.6
	TNR	86.0 ± 12.1	93.4 ± 7.9	90.2 ± 9.5	90.2 ± 9.3	90.6 ± 9.7	92.0 ± 8.0	94.2 ± 7.3
	TPR	97.6 ± 2.5	96.9 ± 2.6	99.0 ± 1.2	99.1 ± 1.1	98.8 ± 1.3	98.9 ± 1.1	98.9 ± 1.3

6 Conclusion

In this paper, an information-based classifier ensemble method based on Dempster-Shafer theory has been developed to monitor the geometrically complex parts in terms of their structural quality by using their vibration responses. The method consists of three main stages: classifier generation, selection, and fusion. In the first stage, several classifiers have been trained. The classifier selection was based on joint mutual information and the "maximum of minimum" concepts. An improved technique based on the Dempster-Shafer theory of evidence was used for classifier fusion. To deal with conflicting evidences, the classifiers'

outputs were optimally redistributed by considering the distance between the predicted and the target output. Further, the proposed method has been equipped with five different weightings among which the best one could be selected by using validation accuracy.

To validate the performance of the proposed method, it has been applied to thirteen UCI and KEEL machine learning datasets. It is also applied to two vibrational datasets generated from one synthetic dataset and one real experimental dataset. The synthetic dataset was obtained from a FE model of dogbone cylinder whereas, the experimental dataset collected from equiax polycrystalline Nickel alloy first-stage turbine blades with complex geometry and different types and severities of damages. The results demonstrated the benefit of employing the proposed ensemble method. By comparing with five state-of-the-art DST-based classifier fusion methods, the better performance of the proposed framework was demonstrated.

7 Acknowledgment

The authors gratefully acknowledge the ICON project DETECT-ION (HBC.2017.0603) which fits in the SIM research program MacroModelMat (M3) coordinated by Siemens (Siemens Digital Industries Software, Belgium) and funded by SIM (Strategic Initiative Materials in Flanders) and VLAIO (Flemish government agency Innovation & Entrepreneurship). The authors acknowledge the ICON project ALMA (HBC.2018.0427) which fits in the research program STREAM funded by SIM (Strategic Initiative Materials in Flanders) and VLAIO (Flemish government agency Innovation & Entrepreneurship). Vibrant Corporation is also gratefully acknowledged for providing anonymous datasets of the turbine blades and dogbone cylinders.

References

[1] A. Thompson, I. Maskery, R.K. Leach, X-ray computed tomography for additive manufacturing : A review, Measurement Science and Technology. 27 (2016) 072001.

[2] A. Du Plessis, I. Yadroitsev, I. Yadroitsava, S.G. Le Roux, X-Ray microcomputed tomography in additive manufacturing: a review of the current technology and applications, 3D Printing and Additive Manufacturing. 5 (2018) 227–247.

[3] E. Adeli, B. Rosić, H.G. Matthies, S. Reinstädler, D. Dinkler, Bayesian Parameter Determination of a CT-Test Described by a Viscoplastic-Damage Model Considering the Model Error, Metals. 10 (2020) 1141.

[4] F. Mevissen, M. Meo, A nonlinear ultrasonic modulation method for crack detection in turbine blades, Aerospace. 7 (2020) 72.

[5] G. Poelman, S. Hedayatrasa, J. Segers, W. Van Paepegem, M. Kersemans, Adaptive spectral band integration in flash thermography: Enhanced defect detectability and quantification in composites, Composites Part B: Engineering. (2020) 108305.

[6] F. Ciampa, P. Mahmoodi, F. Pinto, M. Meo, Recent Advances in Active Infrared Thermography for Non-Destructive Testing of Aerospace Components, Sensors. 18 (2018) 609.

[7] J. V. Heffernan, L. Jauriqui, E. Biedermann, A. Mayes, R. Livings, B. Goodlet, S. Mazdiyasni, Process compensated resonance testing models for quantification of creep damage in single crystal nickel-based superalloys, Materials Evaluation. (2017).

[8] L. Cheng, V. Yaghoubi, W. van Paepegem, M. Kersemans, Mahalanobis classification system (MCS) integrated with binary particle swarm optimization for robust quality classification of complex metallic turbine blades, Mechanical Systems and Signal Processing. 146 (2021) 107060.

[9] E. Todorov, R. Spencer, S. Gleeson, M. Jamshidinia, S.M.K. Ewi, America makes: National additive manufacturing innovation institute (NAMII) project 1: Nondestructive evaluation (NDE) of complex metallic additive manufactured (AM) structures, No. 55028GTH. Edison welding inst inc columbus OH, 2014.

[10] F. Mevissen, M. Meo, A Review of NDT/Structural Health Monitoring Techniques for Hot Gas Components in Gas Turbines, Sensors. 19 (2019) 711.

[11] E. Pomponi, A. Vinogradov, A real-time approach to acoustic emission clustering, Mechanical Systems and Signal Processing. 40 (2013) 791–804.

[12] M. Salucci, N. Anselmi, G. Oliveri, S. Member, P. Calmon, R. Miorelli, C. Reboud, A. Massa, Real-time NDT-NDE through an innovative adaptive partial least squares SVR inversion approach, IEEE Transactions on Geoscience and Remote Sensing. 54 (2016) 6818–6832.

[13] H.J. Lim, H. Sohn, Y. Kim, Data-driven fatigue crack quantification and prognosis using nonlinear ultrasonic modulation, Mechanical Systems and Signal Processing. 109 (2018) 185–195.

[14] S. Sambath, P. Nagaraj, N. Selvakumar, Automatic defect classification in ultrasonic NDT using artificial intelligence, Journal of Nondestructive Evaluation. (2011).

[15] M. Abdelniser, A. Abdulbaset, R. Omar M., Reducing sweeping frequencies in microwave NDT employing machine learning feature selection, Sensors. 16 (2016) 1–14.

[16] T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning: Data Mining, Inference, and Prediction, Springer Science & Business Media, 2009.

[17] S. Rothe, B. Kudszus, D. Söffker, Does classifier fusion improve the overall performance? Numerical analysis of data and fusion method characteristics influencing classifier fusion performance, Entropy. 21 (2019) 866.

[18] Y. Pan, L. Zhang, X. Wu, M.J. Skibniewski, Multi-classifier information fusion in risk analysis, Information Fusion. 60 (2020) 121–136.

[19] Q. He, X. Li, D.W.N. Kim, X. Jia, X. Gu, X. Zhen, L. Zhou, Feasibility study of a multi-criteria decision-making based hierarchical model for multi-modality feature and multi-classifier fusion: Applications in medical prognosis prediction, Information Fusion. 55 (2020) 207–219.

[20] M.P. Ponti, Combining Classifiers: from the creation of ensembles to the decision fusion, in: 24th SIBGRAPI Conference on Graphics, Patterns, and Images Tutorials, 2011: pp. 1–10.

[21] Z.Y. Wang, C. Lu, B. Zhou, Fault diagnosis for rotary machinery with selective ensemble neural networks, Mechanical Systems and Signal Processing. 113 (2018) 112–130. https://doi.org/10.1016/j.ymssp.2017.03.051.

[22] X. Li, H. Jiang, M. Niu, R. Wang, An enhanced selective ensemble deep learning method for rolling bearing fault diagnosis with beetle antennae search algorithm, Mechanical Systems and Signal Processing. 142 (2020) 106752. https://doi.org/10.1016/j.ymssp.2020.106752.

[23] S. Pang, X. Yang, X. Zhang, Y. Sun, Fault diagnosis of rotating machinery components with deep ELM ensemble induced by real-valued output-based diversity metric, Mechanical Systems and Signal Processing. 159 (2021) 107821. https://doi.org/10.1016/j.ymssp.2021.107821.

[24] V. Yaghoubi, L. Cheng, W. van Paepegem, M. Kersemans, A novel multi-classifier information fusion based on Dempster-Shafer theory: application to vibration-based fault detection, Structural Health Monitoring. (2021). https://doi.org/https://doi.org/10.1177/14759217211007130.

[25] B. Quost, M.H. Masson, T. Denœux, Classifier fusion in the Dempster-Shafer framework using optimized t-norm based combination rules, International Journal of Approximate Reasoning. 52 (2011) 353–374.

[26] R. Lysiak, M. Kurzynski, T. Woloszynski, Optimal selection of ensemble classifiers using measures of competence and diversity of base classifiers, Neurocomputing. 126 (2014) 29–35.

[27] R. Assumpção Silva, A.S. Britto, F. Enembreck, R. Sabourin, L.S. Oliveira, Selecting and Combining Classifiers Based on Centrality Measures, in: International Journal on Artificial Intelligence Tools, World Scientific Publishing Co. Pte Ltd, 2020.

[28] R.A. Silva, A. de S. Britto Jr, F. Enembreck, R. Sabourin, L.E.S. Oliveira, CSBF: A static ensemble fusion method based on the centrality score of complex networks, Computational Intelligence. 36 (2020) 522–556.

[29] F.A. Faria, D.C.G. Pedronette, J.A. dos Santos, A. Rocha, R.D.S. Torres, Rank aggregation for pattern classifier selection in remote sensing images, IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing. 7 (2014) 1103–1115.

[30] G. Brown, An Information Theoretic Perspective on Multiple Classifier Systems, in: International Workshop on Multiple Classifier Systems., Springer, Berlin, Heidelberg, 2009: pp. 344–353.

[31] H.J. Kang, S.W. Lee, An information-theoretic strategy for constructing multiple classifier systems, Proceedings - International Conference on Pattern Recognition. 15 (2000) 483–486.

[32] D. Ruta, B. Gabrys, Classifier selection for majority voting, Information Fusion. 6 (2005) 63–81.

[33] Y.S. Huang, K. Liu, C.Y. Suen, The combination of multiple classifiers by a neural network approach, International Journal of Pattern Recognition and Artificial Intelligence. 09 (1995) 579–597.

[34] L.I. Kuncheva, J.C. Bezdek, R.P.W. Duin, Decision templates for multiple classifier fusion: An experimental comparison, Pattern Recognition. 34 (2001) 299–314.

[35] N. Pizzi, W. Pedrycz, Aggregating multiple classification results using fuzzy integration and stochastic feature selection, International Journal of Approximate Reasoning. 51 (2010) 883–894.

[36] G. Rogova, Combining the results of several neural network classifiers, Studies in Fuzziness and Soft Computing. 219 (2008) 683–692.

[37] A. Al-Ani, M. Deriche, A new technique for combining multiple classifiers using the Dempster-Shafer theory of evidence, Journal of Artificial Intelligence Research. 17 (2002) 333–361.

[38] R.R. Yager, On the dempster-shafer framework and new combination rules, Information Sciences. 41 (1987) 93–137.

[39] D. Dubois, H. Prade, A review of fuzzy set aggregation connectives, Information Sciences. 36 (1985) 85–121. https://www.sciencedirect.com/science/article/pii/0020025585900271 (accessed June 17, 2021).

[40] J. Dezert, A. Tchamova, On the Validity of Dempster's Fusion Rule and its Interpretation as a Generalization of Bayesian Fusion Rule, International Journal of Intelligent Systems. 29 (2014) 223–252.

[41] X. Deng, Q. Liu, Y. Deng, S. Mahadevan, An improved method to construct basic probability assignment based on the confusion matrix for classification problem, Information Sciences. 340–341 (2016) 250–261.

[42] J. Qian, · Xingfeng Guo, · Yong Deng, Y. Deng, A novel method for combining conflicting evidences based on information entropy, 46 (2017) 876–888.

[43] F. Xiao, A novel evidence theory and fuzzy preference approach-based multi-sensor data fusion technique for fault diagnosis, Sensors (Switzerland). 17 (2017).

[44] F. Xiao, B. Qin, A weighted combination method for conflicting evidence in multi-sensor data fusion, Sensors (Switzerland). 18 (2018).

[45] L. Fei, J. Xia, Y. Feng, L. Liu, A novel method to determine basic probability assignment in Dempster–Shafer theory and its application in multi-sensor information fusion, International Journal of Distributed Sensor Networks. 15 (2019) 155014771986587.

[46] Z. Wang, J. Gao, R. Wang, Z. Gao, Y. Liang, Fault recognition using an ensemble classifier based on Dempster–Shafer Theory, Pattern Recognition. 99 (2020) 107079.

[47] Z. Liu, Q. Pan, J. Dezert, J.W. Han, Y. He, Classifier Fusion with Contextual Reliability Evaluation, IEEE Transactions on Cybernetics. 48 (2018) 1605–1618.

[48] Z.G. Liu, Y. Liu, J. Dezert, F. Cuzzolin, Evidence Combination Based on Credal Belief Redistribution for Pattern Classification, IEEE Transactions on Fuzzy Systems. 28 (2020) 618–631.

[49] Z.G. Liu, Q. Pan, J. Dezert, A. Martin, Combination of Classifiers With Optimal Weight Based on Evidential Reasoning, IEEE Transactions on Fuzzy Systems. 26 (2018) 1217–1230.

[50] L. Yang, Classifiers selection for ensemble learning based on accuracy and diversity, in: Procedia Engineering, Elsevier, 2011: pp. 4266–4270.

[51] F.A. Faria, J.A. dos Santos, A. Rocha, R.D.S. Torres, A framework for selection and fusion of pattern classifiers in multimedia recognition, Pattern Recognition Letters. 39 (2014) 52–64.

[52] D.M.J. Tax, R.P.W. Duin, Support Vector Data Description, Machine Learning. 54 (2004) 45–66.

[53] M. Bennasar, Y. Hicks, R. Setchi, Feature selection using Joint Mutual Information Maximisation, Expert Systems With Applications. 42 (2015) 8520–8532.

[54] Y. Zhang, Q. Ji, Efficient sensor selection for active information fusion, IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics. 40 (2010) 719–728.

[55] G. Shafer, Dempster's rule of combination, International Journal of Approximate Reasoning. 79 (2016) 26–40.

[56] T. Denœux, Logistic regression, neural networks and Dempster–Shafer theory: A new perspective, Knowledge-Based Systems. 176 (2019) 54–67.

[57] C.R. Parikh, M.J. Pont, N. Barrie Jones, Application of Dempster-Shafer theory in condition monitoring applications: A case study, Pattern Recognition Letters. 22 (2001) 777–785.

[58] L. Xu, A. Krzyżak, C.Y. Suen, Methods of Combining Multiple Classifiers and Their Applications to Handwriting Recognition, IEEE Transactions on Systems, Man and Cybernetics. 22 (1992) 418–435.

[59] F. Xiao, Multi-sensor data fusion based on the belief divergence measure of evidences and the belief entropy, Information Fusion. 46 (2019) 23–32.

[60] D. Dua, C. Graff, UCI Machine Learning Repository, (2019).

[61] A. Fernández, S. García, M.J. del Jesus, F. Herrera, A study of the behaviour of linguistic fuzzy rule based classification systems in the framework of imbalanced data-sets, Fuzzy Sets and Systems. 159 (2008) 2378–2398.

[62] J. Heffernan, E. Biedermann, A. Mayes, R. Livings, L. Jauriqui, S. Mazdiyasni, Validation of process compensated resonance testing (PCRT) sorting modules trained with modeled data, in: AIP Conference Proceedings, 2019.

Appendix I. Example on classifier ranking procedure

This appendix provides a toy example to illustrate the proposed classifier selection procedure. Suppose there are four individual classifiers (C_1, C_2, C_3, and C_4) trained on a two-class classification problem. Given 6 samples with the true class of $\boldsymbol{y}_t = [1\ 1\ 1\ 0\ 0\ 0]$, the classifiers give the output vectors $\hat{\boldsymbol{y}}_1 = [1\ 0\ 1\ 0\ 1\ 0]$, $\hat{\boldsymbol{y}}_2 = [1\ 1\ 1\ 1\ 0\ 0]$, $\hat{\boldsymbol{y}}_3 = [1\ 1\ 0\ 1\ 0\ 1]$, and $\hat{\boldsymbol{y}}_4 = [1\ 1\ 1\ 1\ 1\ 1]$. Now, to obtain the Mutual information, Conditional mutual information, and joint mutual information the following steps should be taken

1- Estimation of individual probability mass functions
   a. $P_t = [0.5, 0.5]$
   b. $P_1 = [0.5, 0.5], P_2 = [0.33, 0.67], P_3 = [0.33, 0.67], P_4 = [0, 1]$
2- Estimation of joint probability mass function

$$P_{1t} = \begin{bmatrix} 0.33 & 0.17 \\ 0.17 & 0.33 \end{bmatrix}, P_{2t} = \begin{bmatrix} 0.33 & 0.00 \\ 0.17 & 0.50 \end{bmatrix}$$
$$P_{3t} = \begin{bmatrix} 0.17 & 0.17 \\ 0.33 & 0.33 \end{bmatrix}, P_{4t} = \begin{bmatrix} 0.00 & 0.00 \\ 0.50 & 0.50 \end{bmatrix} \quad \text{(I-1)}$$

3- Estimation of Mutual information Eq. (12)

$$MI_1 = 0.057, MI_2 = 0.318$$
$$MI_3 = 0.000, MI_4 = 0.000 \quad \text{(I-2)}$$

4- Estimation of joint probability mass functions
   a. with the first classifier

$$P_{11} = \begin{bmatrix} 0.50 & 0.00 \\ 0.00 & 0.50 \end{bmatrix}, P_{21} = \begin{bmatrix} 0.17 & 0.17 \\ 0.33 & 0.33 \end{bmatrix}$$
$$P_{31} = \begin{bmatrix} 0.00 & 0.33 \\ 0.50 & 0.17 \end{bmatrix}, P_{41} = \begin{bmatrix} 0.00 & 0.00 \\ 0.50 & 0.50 \end{bmatrix} \quad \text{(I-3)}$$

   b. 3rd order with the true model and the first classifier on class 0

$$P_{1t1,0} = \begin{bmatrix} 0.33 & 0.17 \\ 0.00 & 0.00 \end{bmatrix}, P_{2t1,0} = \begin{bmatrix} 0.17 & 0.00 \\ 0.17 & 0.17 \end{bmatrix}$$
$$P_{3t1,0} = \begin{bmatrix} 0.00 & 0.00 \\ 0.33 & 0.17 \end{bmatrix}, P_{4t1,0} = \begin{bmatrix} 0.00 & 0.00 \\ 0.33 & 0.17 \end{bmatrix} \quad \text{(I-4)}$$

   c. 3rd order with the true model and the first classifier on class 1

$$P_{1t1,1} = \begin{bmatrix} 0.00 & 0.00 \\ 0.17 & 0.33 \end{bmatrix}, P_{2t1,1} = \begin{bmatrix} 0.17 & 0.00 \\ 0.00 & 0.33 \end{bmatrix} \quad \text{(I-5)}$$

$$P_{3t1,1} = \begin{bmatrix} 0.17 & 0.17 \\ 0.00 & 0.17 \end{bmatrix}, P_{4t1,1} = \begin{bmatrix} 0.00 & 0.00 \\ 0.17 & 0.33 \end{bmatrix}$$

5- Estimation of conditional mutual information Eq. (14)

$$CMI = \begin{bmatrix} 0.00 & 0.41 & 0.09 & 0.00 \\ 0.14 & 0.00 & 0.06 & 0.00 \\ 0.14 & 0.37 & 0.00 & 0.00 \\ 0.06 & 0.32 & 0.00 & 0.00 \end{bmatrix} \quad \text{(I-6)}$$

6- Estimation of joint mutual information Eq. (13)

$$JMI = \begin{bmatrix} 0.06 & 0.46 & 0.14 & 0.06 \\ 0.46 & 0.32 & 0.37 & 0.32 \\ 0.14 & 0.37 & 0.00 & 0.00 \\ 0.06 & 0.32 & 0.00 & 0.00 \end{bmatrix} \quad \text{(I-7)}$$

7- Classifier ranking by JMI

a. $C_1^r, C_2^r = \arg\max JMI = 2,1$

b. $C_3^r = \arg\max(\min(JMI(2,:), JMI(1,:))) = 4$

c. $C^r = [2,1,4,3]$