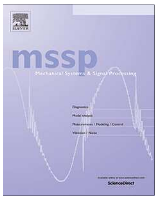
Mechanical Systems and Signal Processing 134 (2019) 106294

Contents lists available at ScienceDirect

Mechanical Systems and Signal Processing

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



Probabilistic active learning: An online framework for structural health monitoring

L.A. Bull *, T.J. Rogers, C. Wickramarachchi, E.J. Cross, K. Worden, N. Dervilis

Dynamics Research Group, Department of Mechanical Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, UK

ARTICLE INFO

Article history:
Received 18 March 2019
Received in revised form 1 July 2019
Accepted 7 August 2019

Keywords:
Damage detection
Pattern recognition
Semi-supervised learning
Structural health monitoring

ABSTRACT

A novel, probabilistic framework for the classification, investigation and labelling of data is suggested as an online strategy for Structural Health Monitoring (SHM). A critical issue for data-based SHM is a lack of descriptive labels (for measured data), which correspond to the condition of the monitored system. For many applications, these labels are costly and/or impractical to obtain, and as a result, conventional supervised learning is not feasible. This fact forces a dependence on outlier analysis, or one-class classifiers, in practical applications, as a means of damage detection. The model suggested in this work, however, allows for the definition of a multi-class classifier, to aid both damage detection and identification, while using a limited number of the most informative labelled data. The algorithm is applied to three datasets in the online setting; the Z24 bridge data, a machining (acoustic emission) dataset, and measurements from ground vibration aircraft tests. In the experiments, active learning is shown to improve the online classification performance for damage detection and classification.

1. Introduction

Structural health monitoring (SHM) systems look to provide a framework for the classification and localisation of damage, following preliminary damage detection. As such, SHM frameworks require the categorisation of many data-groups, i.e. classes, relating to different states of structural health, rather than simply classifying data as either normal or novel (outlier analysis) [1,2]. For engineering datsets, particularly in SHM, a critical issue with the multi-class problem is a lack of comprehensive labelled data, which are required to learn in a (standard) supervised classification algorithm. Furthermore, in an online setting for SHM, the measured data arrive as a stream, incrementally, throughout the lifetime of the monitored structure; as a result, a model should be capable of adapting and updating as new data become available.

Considering these issues, SHM systems should offer three characteristics. Firstly, the system should be adaptive, incorporating any new classes (novel data-groups) as they are discovered; these might relate to damage or various operational conditions. Secondly, a system should be capable of running on-line; that is, the algorithm should be computationally efficient, in order to update and adapt during operation. Finally, the model must be capable of accurate diagnostics (ideally probabilistic, to include the uncertainty of predictions) while only requesting descriptive labels for the most informative measured data; this is critical for engineering applications, as the investigation of any abnormal data is regularly impractical and

* Corresponding author.
E-mail address: l.a.bull@sheffield.ac.uk (L.A. Bull).

https://doi.org/10.1016/j.ymssp.2019.106294
0888-3270/© 2019 The Authors. Published by Elsevier Ltd.

expensive. This paper outlines an approach to address the investigation of engineering data streams following an active learning, probabilistic framework for online SHM.

The layout of the paper is as follows. Section 2 provides an overview of partially-supervised learning, in the context of SHM, including related work. Section 3 defines the probabilistic model, and a framework for guided sampling, which is used to inform damage detection and multi-class classification. Section 4 describes how the model is incorporated into an online SHM strategy. Section 5 demonstrates empirical improvements of active learning, through the application of the heuristic to three datasets. Finally, Sections 6 and 7 suggest future work, and offer concluding remarks.

2. Partially-supervised learning for SHM

SHM involves monitoring an engineering structure or system using observation data, in order to make informed predictions about the current operating, environmental or damage condition. More specifically, when following a data-driven approach, pattern recognition and machine learning tools are applied to learn patterns in the data, in order to inform the current condition of the system. As a result, the classification of measured data via a robust model, learnt using a limited subset of training data, is a fundamental problem. Generally speaking, the i^{th} measured data point, $\boldsymbol{x}_i \in X$, can be categorised according to a descriptive label, $y_i \in Y$, which corresponds to the ground truth of a classification problem. In the context of SHM, the observations $\boldsymbol{x}_i$ would represent features extracted from the raw measurements following signal processing, while the descriptive labels, y_i, are used to inform which groups of measured data relate to different conditions; for example: is the system operating normally, under extreme temperatures, or, most critically, is the system damaged? The key stages within a typical SHM strategy are shown in Fig. 1.

From a probabilistic classification perspective, it is assumed that the features are defined by a random vector in a D-dimensional feature space, such that $\boldsymbol{x}_i \in X$ and $X \in \mathbb{R}^D$. Furthermore, for a discrete classification problem, the descriptive labels are defined by a discrete random variable, such that $y_i \in Y = \{1, \ldots, K\}$. K is the number of classes which define the operational and health conditions, and Y denotes the label space. Typically, two main frameworks are used to learn patterns from data in the context of SHM [1]; these are *unsupervised* and *supervised* learning.

2.1. Traditional pattern recognition in SHM

Supervised pattern-recognition algorithms require fully-labelled training-data, $\mathcal{D}_l$, such that [3],

$$\mathcal{D}_l = \{(\boldsymbol{x}_i, y_i) | \boldsymbol{x}_i \in X, y_i \in Y\}_{i=1}^{n} \tag{1}$$

for n collected data points. As the training-set, $\mathcal{D}_l$, includes both measured data and descriptive labels, a supervised classifier can learn a mapping between the feature space and the label space, $f : X \to Y$. The classifier, f, can then be used to predict the label of future measurements, and thus, make diagnostic decisions in an SHM context. In contrast, *unsupervised* techniques are applied when only the measured data are available to build a model. In this case, the training-set becomes [3],

$$\mathcal{D}_u = \{\tilde{\boldsymbol{x}}_i | \tilde{\boldsymbol{x}}_i \in X\}_{i=1}^{m} \tag{2}$$

$\tilde{\boldsymbol{x}}_i$ is used to denote the measured data that are unlabelled. A variety of data analysis and machine learning tools can be applied to unlabelled datasets, $\mathcal{D}_u$. Some examples include: dimensionality reduction, novelty detection, outlier analysis and clustering [4]. These techniques aim to find patterns within a dataset from the information contained within the measured observations alone. As a result, the learning process must be informed by a cost function that does not utilise any of the information from the label space, Y, as this information is not available [3].

The unsupervised setting is relevant in an engineering context, as comprehensive labels to describe the measured data are rarely available [1]. For example, in order to define a *complete* labelled SHM dataset, the system must be measured across all operational and damaged conditions, while the structure is regularly inspected by an engineer to annotate the measured data. Additionally, a dataset recorded from one structure is not necessarily relevant to another (nominally) identical system. Therefore, traditional supervised learning of high-value systems (such as aerospace or civil structures) is clearly impractical/infeasible. Currently, this fact forces a dependence on traditional unsupervised techniques in many practical applications; specifically, novelty detection.

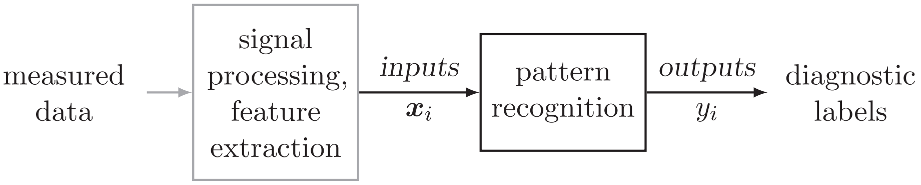


Fig. 1. Flow chart to illustrate SHM strategy definitions.

2.2. Partially-supervised learning

An alternative approach, however, is to apply *partially-supervised* pattern recognition [3]; these algorithms make use of both labelled data, $\mathcal{D}_l$, and unlabelled data, $\mathcal{D}_u$, such that dataset used by the algorithm is,

$$\mathcal{D} = \mathcal{D}_l \cup \mathcal{D}_u \tag{3}$$

Consequently, partially-supervised techniques can make use of a limited subset of labelled data, when annotation by an engineer proves to be impractical/expensive. Two of the main techniques within the partially-supervised learning family are *semi-supervised* and *active* learning algorithms. Briefly, *semi-supervised* learning utilises both the labelled and unlabelled data to inform the classification mapping, $f : X \mapsto Y$. In other words, a semi-supervised learner uses the available unlabelled data $\mathcal{D}_u$, to further constrain a supervised classifier, learnt from the labelled data $\mathcal{D}_l$, to improve the classification performance. Unlabelled data can be incorporated in various ways. The most simple approach, self-labelling [5,6], trains a classifier using $\mathcal{D}_l$, and then predicts the labels for unlabelled instances $\tilde{\boldsymbol{x}}_i$. Finally, the classifier is retrained using the complete dataset, $\mathcal{D}$. In this case, some labels in $\mathcal{D}$ are the ground truth, from the supervised dataset, and others are assumed based on the label predictions. This approach to semi-supervised learning has been shown to improve the predictive performance of damage classification in [6]. The focus of this work, however, considers *active learning* variants of partially-supervised learning.

2.2.1. Active learning

Active learning is another form of partially-supervised learning [3]. As with semi-supervised techniques, active algorithms will make use of both $\mathcal{D}_l$ and $\mathcal{D}_u$; however, an active learner will query/annotate unlabelled data in $\mathcal{D}_u$ to automatically extend the labelled dataset, $\mathcal{D}_l$, in an intelligent and adaptive manner. The generalised (and simplified) active learning framework is illustrated in Fig. 2.

Active algorithms can be applied offline to a large pool of collected data [7], or online, to drifting data streams (which evolve through time) [8]. In the online setting, if an algorithm can adapt and update, while only requesting critical labels, this is highly significant to data-based SHM. For example, if the measured data are recorded live from a wind-turbine 80 km off-shore, any novel data that might relate to damage would potentially need to be investigated manually. This requires an engineer to travel to the wind-turbine by air or sea, and then inspect the structure to explain any inconsistencies observed in the measured data. If a statistical model can be used to determine when only the most informative/critical observations need to be investigated, this can lead to significant reductions in maintenance costs.

2.3. Related work

In the context of data-based SHM, there has been a growing interest in partially-supervised methods [6,9,10], as an algorithm that is semi-supervised and active can bring several advantages [6]. Active learning has been applied to SHM data in previous work [6], using Dasgupta and Hsu's tree-based, nonparametric algorithm [11]. This model also utilises automated querying to build an informative training-set; however, in this application, a large pool of the measured data are (generally) required *a priori* to build the tree structure, which is used to inform and direct guided sampling. As a result, while the algorithm provides a significant improvement to the classification, the heuristic applied in [6] is less suitable for online SHM, where the data arrives incrementally. In another paper [9], an online SHM strategy is built using an unsupervised clustering algorithm, with the view to building a partially-supervised SHM strategy. The approach suggested in [9] differs from the strategy suggested in this work, as the Dirichlet process is unsupervised; therefore, labelled data are not used at the algorithm level. However, important concepts from the framework suggested in [9] can be combined with the tools suggested in this work. Finally, the use of semi-supervised learning has been applied to civil infrastructure datasets in the context of SHM [12]. Again, while this work is related (as it concerns partially-supervised learning), it focusses on semi-supervised learning, rather than active methods. Specifically, the model suggested in this work defines a novel tool for multi-class classification in the context of *online* SHM, such that the labelling of data (and thus the investigation of the system) is limited and directed via a probabilistic active learning framework.

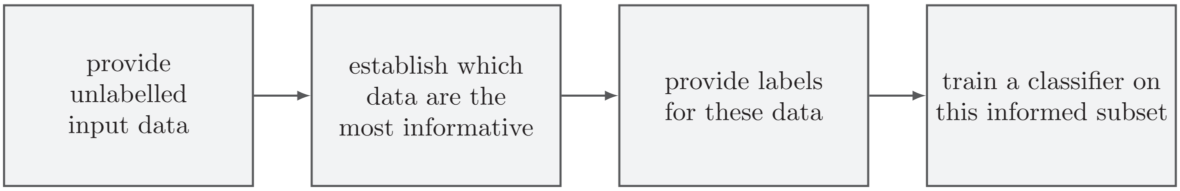

provide unlabelled input data
establish which data are the most informative
provide labels for these data
train a classifier on this informed subset

Fig. 2. The general active learning heuristic.

3. A probabilistic model for guided sampling

A probabilistic approach is suggested as the foundation for an active framework with engineering data. This approach is built around a supervised probabilistic mixture model, which is learnt from a small initial (random) sample of labelled measured data. As with existing models in the literature [4,13,14], the measured data, $\boldsymbol{x}_i$, are assumed to be sampled from a parametric mixture model; specifically, a mixture of K Gaussian distributions, such that each class, $Y = \{1, \ldots, K\}$, is generated by a multi-variate Gaussian distribution,

$$p(\boldsymbol{x}_i | y_i = k) = \mathcal{N}(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) \tag{4}$$

A pair of parameters (mean, $\boldsymbol{\mu}_k$, and covariance, $\boldsymbol{\Sigma}_k$) are used to define the distribution of $\boldsymbol{x}_i$ for each class in Y. Therefore, there are K parameter sets used to describe the mixture model over the feature space, $\{(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1), \ldots, (\boldsymbol{\mu}_K, \boldsymbol{\Sigma}_K)\}$.

As discussed, the labels, y_i, define the current operating or health condition. This model involves classifying data, therefore, the labels are discrete, and it is appropriate to assume they are categorically distributed [14],

$$p(y_i) = Cat(\boldsymbol{\lambda}) \tag{5}$$

where $\boldsymbol{\lambda} = \{\lambda_1, \ldots, \lambda_K\}$ are the *mixing proportions* for each class $k \in Y$, such that,

$$P(y_i = k) = \lambda_k, \forall k \in Y \tag{6}$$

In words, this states that the probability of the label y_i being equal to the value for class k is λ_k, where $\sum_{k \in Y} \lambda_k = 1$. The parameters $\boldsymbol{\lambda}$ complete the set of parameters used to describe the generative statistical model, $p(y_i, \boldsymbol{x}_i)$,

$$\Theta = \{(\boldsymbol{\mu}_1, \boldsymbol{\Sigma}_1, \lambda_1), \ldots, (\boldsymbol{\mu}_K, \boldsymbol{\Sigma}_K, \lambda_K)\} \tag{7}$$

3.1. A Bayesian approach

The most straight forward estimate of the model parameters, Θ, is the maximum likelihood estimate given the available data $\mathcal{D}_l$. In this case, Θ corresponds to the sample mean and covariance, and the sample mixing parameters. While a maximum likelihood approach is intuitive, it can be poorly representative of the underlying distribution of the data when the sample size, n, is small [4]. For example, consider a class of data which relates to one of the permitted operating conditions of a system; these data could represent the normal operation of a bridge during cold temperatures. Although an engineer might expect this behavior to occur frequently during winter, it may have been observed infrequently in the current dataset $\mathcal{D}_l$. In this case, the maximum likelihood estimate would predict an unreasonably low probability (i.e. mixing proportion) for that class, as the parameters have been defined such that only the available data is the most likely. In other words, the model has *overfit* the training data, and this can lead to poor generalisation when predicting new data.

To prevent over-training and generalisation issues, various methods can be applied to regularise or validate a maximum likelihood model [15]. Alternatively, a Bayesian approach can address the issue of overtraining; this can be interpreted as a form of self-regularisation. In this case, the parameters of the model, Θ, are also considered to be random variables, and prior knowledge is incorporated to provide a more robust estimate of the model. For further details behind the motivations of a Bayesian approach, refer to [14,16].

Considering the distribution of the measured data over the feature space, X, a prior is placed over the mean and covariance parameters of each class, $\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k$. A natural choice of prior, which is conjugate to the Gaussian distribution (leading to analytically tractable solutions) is the Normal-inverse-Wishart (NIW) distribution [4],

$$p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = NIW(\boldsymbol{m}_0, \kappa_0, \nu_0, \boldsymbol{S}_0) \tag{8}$$

The hyperparameters of the mixture model $(\boldsymbol{m}_0, \kappa_0, \nu_0, \boldsymbol{S}_0)$ can be interpreted as follows: $\boldsymbol{m}_0$ is the prior mean for the location of each class $\boldsymbol{\mu}_k$, and κ_0 determines the strength of the prior [4]; $\boldsymbol{S}_0$ is (proportional to) the prior mean of the covariance, $\boldsymbol{\Sigma}_k$, and ν_0 determines the strength of that prior [4]. These hyperparemeters are defined such that the prior belief states that each class is represented by a zero-mean and unit-variance Gaussian distribution. (Specifically, $p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = NIW(\boldsymbol{0}, 1, D, \mathbb{I})$, where $\mathbb{I}$ is the identity matrix $[D \times D]$, and $\boldsymbol{0}$ is a D-dimensional vector of zeros.) In other words, the prior assumes that the input data are normalised in the feature-space, and as such, the measured data are normalised within the online heuristic, to support this belief.

Considering the distribution over the label space, Y, a Dirichlet prior is placed over the mixing proportions [14], $\boldsymbol{\lambda}$,

$$p(\boldsymbol{\lambda}) = Dir(\boldsymbol{\alpha}) \propto \prod_{k=1}^{K} \lambda_k^{\alpha_k - 1} \tag{9}$$

Again, this is a natural choice in prior, as the Dirichlet distribution is conjugate to the categorical distribution [14].

This introduces the hyperparameters, $\boldsymbol{\alpha} = \{\alpha_1, \ldots, \alpha_K\}$, which can be used to incorporate any prior belief of the probability (or weighting) of each class. In this application, each class is assumed to be *equally* weighted, such that $\alpha_k = n/K, \forall k$. This prior is used as it represents a general case; if (application specific) prior-knowledge of the class weights is available, it should be included. The generative statistical model, $p(y_i, \boldsymbol{x}_i, \Theta)$, has now been defined. The graphical model corresponding to the problem (including dependences) is shown in Fig. 3, including any hyperparameters.

The set of labelled data, $\mathcal{D}_l$, is used to establish the initial number of classes, K, and split the measured data into groups according to their label. These data can then be used to calculate the Bayesian estimates of the model parameters. (Note, in the context of SHM, the *initial* measured data are regularly assumed to represent a single class, i.e. $K = 1$. These measurements should, hopefully, relate to the normal-operating-condition only.) As conjugate prior distributions have been assumed, the posterior distribution over the parameter estimates can be found analytically; these are calculated for each class, $k \in Y$. Firstly, the posterior distribution of $(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k)$ is NIW, with updated parameters [4] (denoted by subscript n),

$$p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k | y_i = k, \mathcal{D}_l) = NIW(\boldsymbol{m}_n, \kappa_n, \nu_n, \boldsymbol{S}_n) \tag{10}$$

such that [4],

$$\kappa_n = \kappa_0 + n_y, \tag{11a}$$

$$\boldsymbol{m}_n = \frac{\kappa_0}{\kappa_0 + n_k}\boldsymbol{m}_0 + \frac{n_k}{\kappa_0 + n_k}\bar{\boldsymbol{x}}_k, \tag{11b}$$

$$\nu_n = \nu_0 + n_k, \tag{11c}$$

$$\boldsymbol{S}_n = \boldsymbol{S}_0 + \boldsymbol{S} + \kappa_0 \boldsymbol{m}_0 \boldsymbol{m}_0^\top - \kappa_n \boldsymbol{m}_n \boldsymbol{m}_n^\top. \tag{11d}$$

n_k is the count (number) of observations in $\mathcal{D}_l$ with the label k, and $\bar{\boldsymbol{x}}_k$ is the sample mean of the observations with the label k. The uncentered sum-of-squares matrix for the observations in class k is $\boldsymbol{S} = \sum_{i=1}^{n_y} \boldsymbol{x}_i \boldsymbol{x}_i^\top$. This result has an intuitive interpretation. The posterior mean $\boldsymbol{m}_n$ is a convex combination of the prior mean, $\boldsymbol{m}_0$, and the maximum likelihood estimate (the sample mean), with 'strength' $\kappa_0 + n_y$ [4]. The posterior scatter matrix $\boldsymbol{S}_n$ is the prior scatter matrix $\boldsymbol{S}_0$, plus the empirical scatter matrix, plus an extra term due to the uncertainty associated with the mean [4].

Similarly, the posterior for the parameters of the categorical distribution over Y is Dirichlet [14],

$$p(\boldsymbol{\lambda} | \mathcal{D}_l) \propto \prod_{y=1}^{K} \lambda_y^{n_y + \alpha_y - 1}. \tag{12}$$

In order to make class predictions for the unlabelled data, $\tilde{\boldsymbol{x}}_i \in \mathcal{D}_u$, the posterior predictive distributions over the labels, Y, and the observations, X, can be found analytically. This is done by marginalising out the parameters from the model [4,14]. For unlabelled measurements, $\tilde{\boldsymbol{x}}_i \in X$, the posterior predictive distribution is a Student-t distribution [4],

$$p(\tilde{\boldsymbol{x}}_i | y_i = k, \mathcal{D}_l) = \int \int p(\tilde{\boldsymbol{x}}_i | \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k | y_i = k, \mathcal{D}_l) d\boldsymbol{\mu}_k d\boldsymbol{\Sigma}_k \tag{13}$$

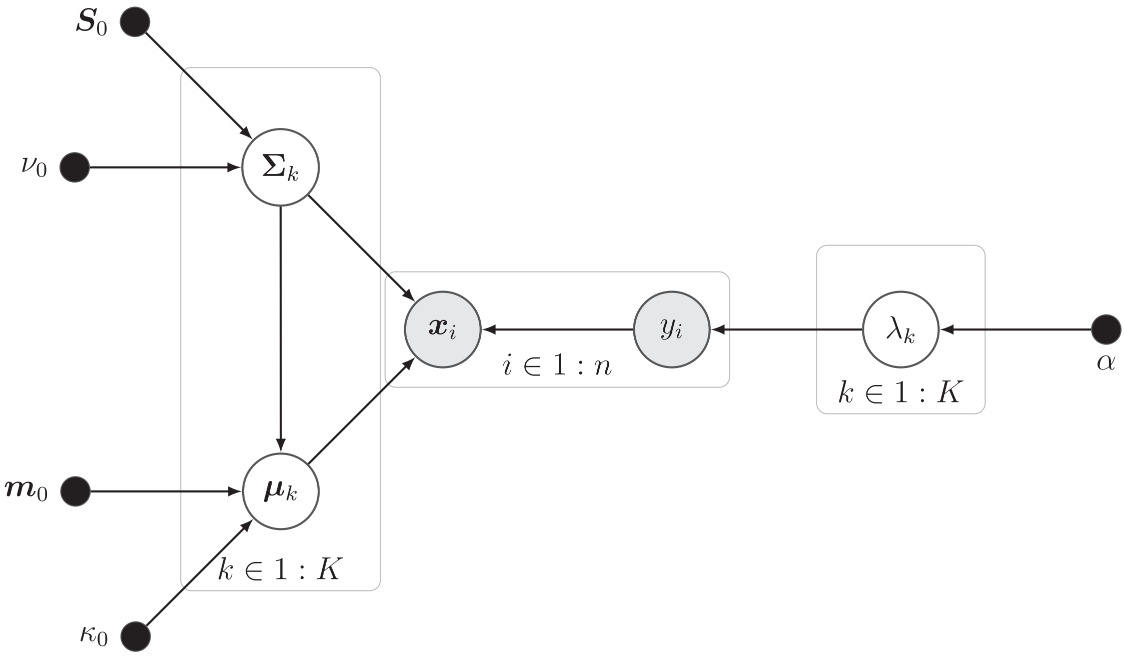


Fig. 3. The probabilistic graphical model of the generative classifier. Shaded and white nodes are the observed and latent variables respectively; arrows represent conditional dependencies; dots represent constants (hyperparameters).

$$= \mathcal{T}\left(\boldsymbol{m}_n, \frac{\kappa_n + 1}{\kappa_n(\nu_n - D + 1)}\boldsymbol{S}_n, \nu_n - D + 1\right). \tag{14}$$

The functional form of the Student-*t* can be found in [4]. The first two terms in Eq. (14) define the mean and scale parameters respectively, and the third term $(\nu - D + 1)$ is the *degrees of freedom*. The Student-*t* distribution is suitable, as it has heavier tails than the Gaussian distribution, to account for the fact that the parameters are estimated from a finite set. However, as more data become available, and the degrees of freedom increase ($n_k \to \infty$, thus $\nu_n \to \infty$), the Student-*t* rapidly approaches the Gaussian distribution.

Similarly, the posterior predictive distribution over the labels, Y, is,

$$p(\tilde{y}_i|\mathcal{D}_l) = \int p(\tilde{y}_i|\boldsymbol{\lambda})p(\boldsymbol{\lambda}|\mathcal{D}_l)d\boldsymbol{\lambda}, \tag{15}$$

$$p(\tilde{y}_i = k|\mathcal{D}_l) = \frac{n_k + \alpha_k}{n + \alpha_0}, \tag{16}$$

where $\alpha_0 = \sum_{k=1}^{K}\alpha_k$ [4].

By utilising the posterior predictive distributions in Eqs. (14) and (16), a generative classifier can be defined using Bayes' rule [4]. This is then used to predict the label distribution, $p(\tilde{y}_i|\tilde{\boldsymbol{x}}_i, \mathcal{D}_l)$, for the *unlabelled* data, $\tilde{\boldsymbol{x}}_i \in \mathcal{D}_u$,

$$p(\tilde{y}_i = k|\tilde{\boldsymbol{x}}_i, \mathcal{D}_l) = \frac{p(\tilde{\boldsymbol{x}}_i|\tilde{y}_i = k, \mathcal{D}_l)p(\tilde{y}_i = k|\mathcal{D}_l)}{p(\tilde{\boldsymbol{x}}_i|\mathcal{D}_l)}, \tag{17}$$

When predicting the label of future data, the maximum *a posteriori* estimate of the class labels is used. This is the value in Y with the highest probability given the observation $\tilde{\boldsymbol{x}}_i$[4], denoted by $\hat{y}_i$,

$$\hat{y}_i = \underset{k \in Y}{\operatorname{argmax}}\,[p(\tilde{y}_i = k|\tilde{\boldsymbol{x}}_i, \mathcal{D}_l)] \tag{18}$$

The marginal likelihood in Eq. (17), which normalises the predictive distribution over Y, is determined by the following integral, which is a discrete sum for a discrete random variable,

$$\begin{aligned} p(\tilde{\boldsymbol{x}}_i|\mathcal{D}_l) &= \int p(\tilde{\boldsymbol{x}}_i|\tilde{y}_i = k, \mathcal{D}_l)p(\tilde{y}_i = k|\mathcal{D}_l)dy(19\text{a}) \\ &\equiv \sum_{k=1}^{K} p(\tilde{\boldsymbol{x}}_i|\tilde{y}_i = k, \mathcal{D}_l)p(\tilde{y}_i = k|\mathcal{D}_l)(19\text{b}) \end{aligned}$$

In summary, a generative classifier has been defined via a *supervised* Gaussian mixture model, with Bayesian estimates of the model parameters. As such, each class of data is represented by a Student-*t* distribution in the feature space, which tends to a Gaussian distribution as more data (in that class) become available. The model is illustrated in the feature space in the next section; additionally, code for the classifier is available on GitHub:https://github.com/labull/probabilistic_active_learning_GMM.

3.1.1. A visual example: acoustic emission data

In order to visualise the mixture model beyond the graphical representation in Fig. 3, the model parameters are learnt for an acoustic emission (AE) dataset in two-dimensions. As these AE data are only used to visualise the model and method, the reader is referred to [17,18] for information regarding the experiment, data collection, signal processing and feature extraction. Briefly, these data represent a two-dimensional, 3-class classification problem, such that $\boldsymbol{x}_i \in \mathbb{R}^2$ and $y_i \in Y = \{1, 2, 3\}$. Each observation, $\boldsymbol{x}_i$, represents the first two principal components [1] of the features extracted from AE burst signals, collected during experiments concerning the box girder of a bridge [18]. The signals are generated by various AE sources, specifically [17]:

- class 1 – frictional processes other than crack related events (clamping in the experimental setup),
- class 2 – crack related events (crack extension and crack-face rubbing),
- class 3 – crack related events, at a distance from the sensor (i.e. AE burst signals with a relatively long rise-time).

A small subset of labelled data (i.e. $\mathcal{D}_l$) is illustrated in Fig. 4a, along with a larger set of unlabelled data, $\mathcal{D}_u$. The mixture model is then learnt using the labelled dataset and label predictions are made for the unlabelled data. The maximum *a posteriori* (MAP) estimate of the parameters of the mixture model are shown in Fig. 4b.

Various probabilistic measures can now be used to estimate which of the measurements in $\mathcal{D}_u$ are the most informative when labelled. These observations can be queried, and the cause can be investigated by the engineer to provide descriptive labels. Following the investigation and labelling of any queried data, $\mathcal{D}_l$ now includes the new observations. Therefore, the model is retrained and then further data can be queried; this process iterates until a label budget is reached, or applied

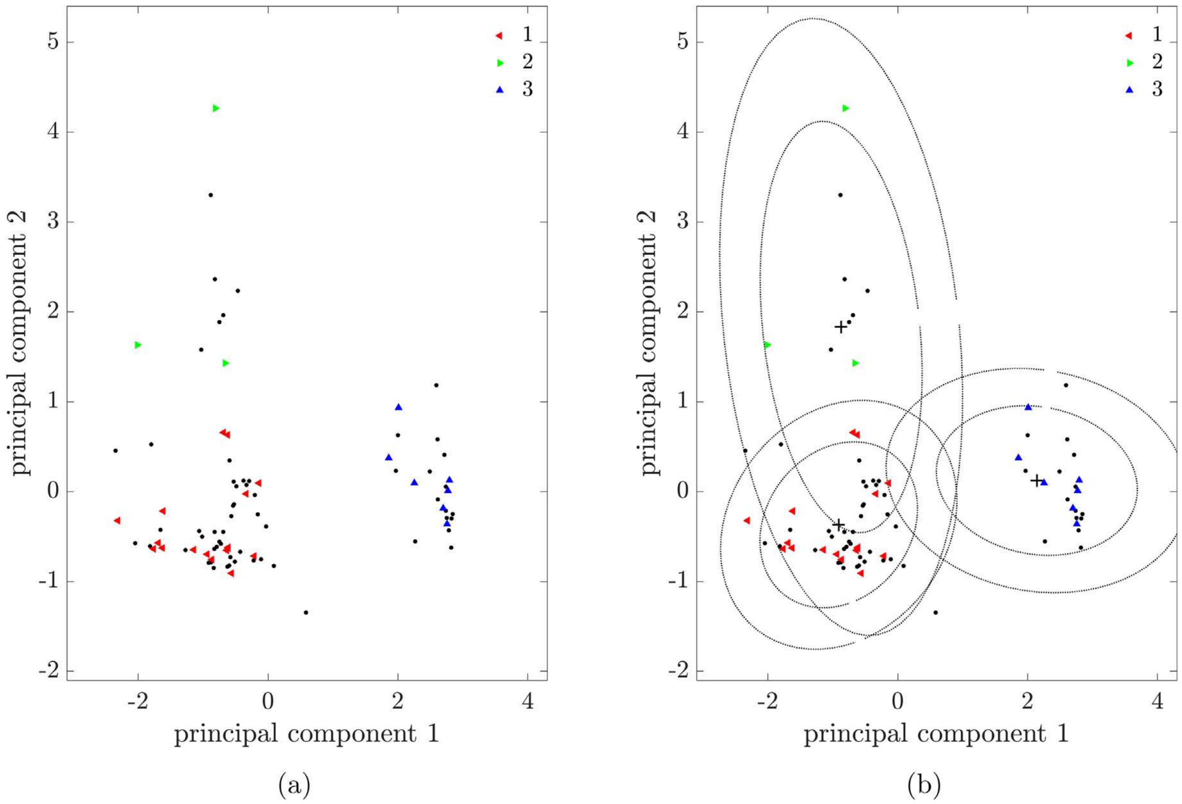


Fig. 4. Multi-class classification of the AE data. (a) Observations in the feature space, X, illustrating the labelled set $\mathcal{D}_l$ (colour markers) and the unlabelled data $\mathcal{D}_u$ (black markers). (b) The generative mixture model $p(\mathbf{x}_i, y_i, \Theta)$; maximum *a posteriori* (MAP) estimate of the mean (+) and covariance (dotted lines represent 2 and 3 sigma). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

sequentially to streaming data (online). This sampling and training framework is typical of *classifier-based* active learning [10,19,20]. Details of the application specific heuristic are provided in the following sections.

3.2. Data query measures — uncertainty sampling

In the active learning literature, there are numerous approaches to define which of the unlabelled data are the most informative [11,19–21]. Generally speaking, if labelled, these data provide the largest increase in the classification performance. However, if queries are too focussed on a specific definition of 'informative', the training-set built by the algorithm can be poorly representative of the underlying distribution of the data; this phenomenon is referred to as *sampling bias*[11]. To combat sampling bias, the query framework should not focus too much on specific regions of the feature-space; this can be achieved by combining several different definitions of 'informative' [20]. Usually, these measures correspond to *representative* or *uncertain* observations, according to the current estimate/model of the underlying data distribution [20,21]. In this work, two probabilistic measures are utilised to direct queries; the typical data queried by these measures are illustrated with the AE data in Fig. 5.

Firstly, the *entropy* of the (categorical) label distribution, $p(\tilde{y}_i = k|\tilde{\mathbf{x}}_i, \mathcal{D}_l)$, can be interpreted as a measure of uncertainty [16]; specifically, the entropy of the outcome $k \in Y$, is defined as the average Shannon information content [16],

$$H(\tilde{y}_i) = -\sum_{k=1}^{K} p(\tilde{y}_i = k|\tilde{\mathbf{x}}_i, \mathcal{D}_l) \log p(\tilde{y}_i = k|\tilde{\mathbf{x}}_i, \mathcal{D}_l). \tag{20}$$

As a result, selecting data from $\mathcal{D}_u$ with a large entropy can be considered uncertainty sampling; that is, selecting data from the unlabelled pool with the most 'mixed' or 'conflicted' label predictions. This criterion will almost always query observations at the boundaries between two or more classes; to demonstrate this, queries directed by a large entropy are illustrated in Fig. 5a. Note, conversely, prioritising low entropy can select measurements near the centre of the data-groups associated with each cluster, i.e. the *representative* examples.

Alternatively, observations in $\mathcal{D}_u$ with the *lowest likelihood* given the current model estimate can be queried, $p(\tilde{\mathbf{x}}_i|\mathcal{D}_l)$. This refers to the marginal likelihood (Eq. (19)) from the Bayes classifier, defined in Eq. (17), i.e.

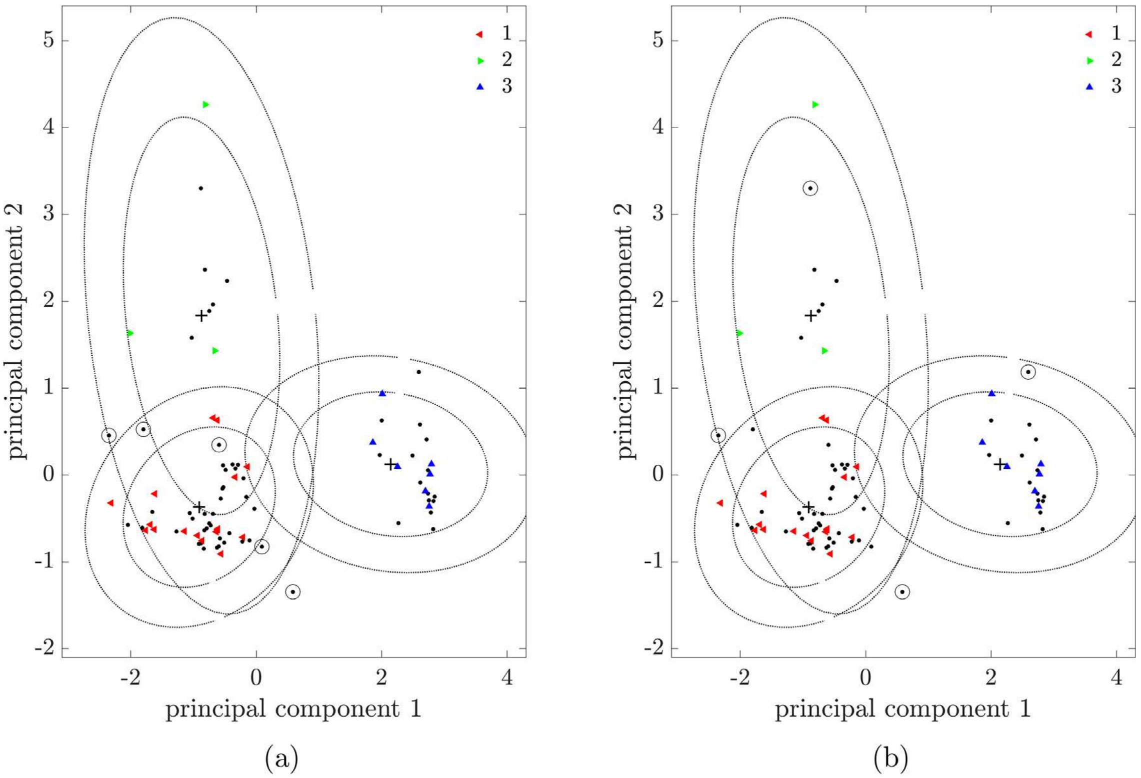


Fig. 5. Queries over the mixture model for the AE data. The labelled set $\mathcal{D}_l$ is shown by the colour markers, and the unlabelled data, $\mathcal{D}_u$, are shown by black markers. The queried data from $\mathcal{D}_u$ are circled; in (a) these data have the *largest entropy*; in (b) the data have the *lowest likelihood* given the current model. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

$$p(\tilde{\mathbf{x}}_i|\mathcal{D}_l) = \sum_{k=1}^{K} p(\tilde{\mathbf{x}}|\tilde{y}_i = k, \mathcal{D}_l)p(\tilde{y}_i = k|\mathcal{D}_l) \tag{21}$$

This can be interpreted as the likelihood of a new observation, having marginalised out the effects of the parameters, Θ, in Eqs. (15) and (13), and the labels, y_i, Eq. (19). Again, querying data with a low likelihood can be seen as uncertainty sampling; however, in this case, the corresponding label distribution is not necessarily 'mixed'. Therefore, the queried data can appear in the cluster extremities that are *not* at the boundary between two or more classes. In other words, these outlying measurements are not necessarily uncertain *in terms of the labels*. Considering these properties, low likelihood data become suitable for querying drifting data streams, typical to online SHM, where the novel data are unlikely to appear between the boundaries of existing classes. Instead, new classes of data are likely to appear as extreme values under the current mixture model, as illustrated in Fig. 5b.

3.2.1. The dangers of active learning

While active learning has been shown to bring significant empirical advantages to pattern recognition models in the literature [11,19–21], the authors wish to reiterate that there are times when selecting training data by a given measure (uncertainty or otherwise) can be worse than random sampling. Specifically, the assumption of most classifiers, and data-based models in general, is that the training data are representative of the underlying data distribution; this implies that the samples are drawn i.i.d from the underlying probability density [19]. While the underlying dataset remains i.i.d in active learning, the samples that define the training set are *guided*; therefore, the data used to learn the algorithm are inherently *not* i.i.d. As a result, care must be taken to ensure that the model does not become misrepresentative. For this reason, it is critical that any application of active learning to engineering data should consider the type (complexity) of data that is being analysed, the quantity of data that is available, and the query budget; as shown in the experiments, the benefits of active learning can vary from dataset to dataset.

4. An online SHM framework

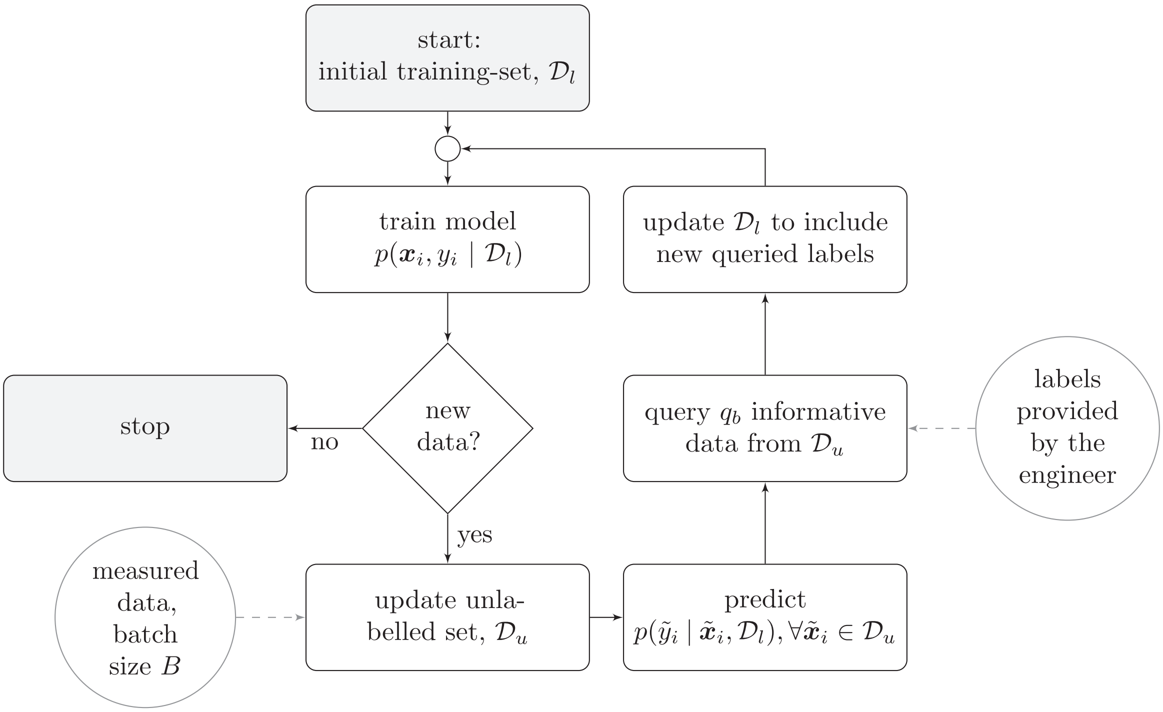
To apply active learning to streaming data for online SHM, a framework for querying data and retraining the model must be formalised. There are various ways to approach this problem in the machine learning literature; for example, query by committee methods [8,19] learn multiple classifiers which can be applied to *drifting data streams*. Disagreement amongst the classifiers is used to direct queries to aid uncertainty sampling [8]. In this work, however, the heuristic is built around a single model [11]. The suggested heuristic is online, despite completely retraining the model (brute-force updates) for each new set of data. Specifically, brute-force learning is possible, as the model is quick to compute, since the parameters are defined through conjugate updates. Furthermore, if desired, the algorithm can be modified to perform online updates of the parameters, mitigating the need to completely 'retrain' [22].

4.1. Guided sampling

In the experiments, the data arrive in batches of size B, and the learner is permitted a limited number of queries per batch, q_b. The number of queries per batch defines the overall sample budget; this can be predefined according to the application and the costs associated with labelling. The initial distribution of data $p(\boldsymbol{x}_i, y_i|\mathcal{D}_l)$ is learnt from the first batch, which is assumed to be wholly labelled as class 1; that is, the normal operating condition. This assumption is reasonable in the context of SHM, as the system should be operating correctly for a large portion of the initial measured data. As a result, this model initialises as a one-class classifier [2]. If a new class of data is discovered following queries, the model updates accordingly; as such, the number of classes K does not need to be defined *a priori*.

The suggested active learner assumes the most informative data are defined through uncertainty sampling, using *entropy* (Eq. (20)) and *marginal likelihood* measures (Eq. (21)). Although this risks sampling bias, as only uncertain samples are targeted, these measurements are assumed to provide the largest increase in classification performance for the experiments in this work (as is common practice in the active learning literature [19]). To address sampling bias to some extent, high-entropy and low-likelihood are *both* considered as measures of *uncertainty*. As discussed, this implies that queries occur in the cluster extremities, *as well as* the boundaries between existing classes. Therefore, sampling a variety of uncertain data in this way should provided an informative training-set, $\mathcal{D}_l$, from the unlabelled streaming data, $\mathcal{D}_u$.

As each new batch of measured data arrives, the model makes a prediction for the unlabelled data $\mathcal{D}_u$, based on the labelled data seen so far in $\mathcal{D}_l$. Note, the dataset $\mathcal{D}_u$ includes the *new* batch, as well as unlabelled data from *previous* batches. The learner then queries q_b measurements from $\mathcal{D}_u$, such that $q_b/2$ records are queried according to high-entropy, and $q_b/2$ are queried with the lowest likelihood. This effectively introduces two hyperparameters: one which determines how many of the data will be labelled, and one which determines what fraction of high-entropy and low-likelihood data should be queried. In this work, an equal number of each measure is queried for simplicity. The sample budget, q_b, is the independent variable in the experiments; therefore, the proportion of each query measure is kept consistent. The investigation of various sampling regimes is being considered for future work. The online heuristic is illustrated in Fig. 6.



start:
initial training-set, $\mathcal{D}_l$
train model
$p(\boldsymbol{x}_i, y_i \mid \mathcal{D}_l)$
update $\mathcal{D}_l$ to include new queried labels
stop
no
new data?
yes
query q_b informative data from $\mathcal{D}_u$
labels provided by the engineer
measured data, batch size B
update unlabelled set, $\mathcal{D}_u$
predict
$p(\tilde{y}_i \mid \tilde{\boldsymbol{x}}_i, \mathcal{D}_l), \forall \tilde{\boldsymbol{x}}_i \in \mathcal{D}_u$

Fig. 6. Flow chart to illustrate the online active learning process.

In order to assess the diagnostic performance of the learner, the full dataset is split in half, using every other sample. This provides a distinct 'moving' test set, $\{\boldsymbol{x}_i^*, y_i^*\}$. The model can then be used to predict the labels for the test data, $\hat{y}_i$ (Eq. (18)), and these can be compared to the actual labels, y_i^*, to determine an online performance metric. The *macrof*$_1$-score is used, which is a weighted balance of precision (P) and recall (R). Precision and recall can be defined in terms of true positives (TP), false positives (FP) and false negatives (FN) for each class, $k \in Y$ [4],

$$P_k = \frac{TP_k}{TP_k + FP_k} \tag{22a}$$

$$R_k = \frac{TP_k}{TP_k + FN_k} \tag{22b}$$

The macro f_1-score is then defined by [4],

$$f_{1,k} = \frac{2P_kR_k}{P_k + R_k} \tag{23a}$$

$$f_{1macro} = \frac{1}{K}\sum_{k \in Y} f_{1,k} \tag{23b}$$

The macro-averaged f_1 metric is used, as this weights the score for each class equally, irrespective of the proportion of the data in each class. This is suitable in the context of online SHM, as newly-discovered groups of data are assumed to be equally important to the classification, despite infrequent observations; i.e. the new data might relate to damage.

5. Experiments

The new heuristic is applied here to three datasets to demonstrate the advantages of active learning for online SHM. To highlight the effects of active learning, the classifier trained using uncertainty sampling is compared to the *same* classifier learnt using data sampled at random from each batch, i.e. standard *passive learning*. As such, for the passive learning benchmark, q_b data are sampled randomly from $\mathcal{D}_u$ for each batch, rather than selecting uncertain data with maximum entropy and the lowest likelihood.

It is important to note — if the active learner queries any *past* data (this is particularly likely with entropy) this may have limitations in practice, as labelling engineering data in hindsight may not be possible, particularly when manual inspection is involved. Intuitively, the structure (or damage) will have changed since that data record. However, in the experiments here, labelling past data is considered to be feasible, as labelling in hindsight can be possible using engineering judgement and other sources of measured data. For example, consider that it is possible to assume that previous outlying data are the result of cold temperature effects, following inspection of temperature plots (as is done with the Z24 data in the next section). The practical limitation of labelling of past data is highlighted, however, as it is an important consideration when applying the heuristic.

5.1. Z24 bridge data

The Z24 bridge was a concrete highway bridge in Switzerland, connecting Koppigen and Utzenstorf. In the late 1990s, before its demolition, it was used for experimental SHM purposes under the SIMCES project [23]. Over a twelve-month time period, a series of sensors were used to capture dynamic response measurements, in order to extract the first four natural frequencies of the structure. Environmental measurements were also recorded, including air temperature, deck temperature, humidity and wind speed [24]. This is a relatively large dataset, with 3932 observations in total. During the benchmark project, different types of damage were artificially introduced towards the end of the monitoring year, starting from observation 3476 [25]. The natural frequencies, as well as deck temperature, are shown in Fig. 7. Fig. 7a illustrates visible fluctuations in the natural frequencies between observations 1200 and 1500, while there is little variation following the introduction of damage at observation 3476. The early fluctuations appear to relate to periods of very low temperature in the bridge deck, which can be observed in the temperature plot, Fig. 7b. It is believed that the asphalt layer in the deck experienced very low temperatures during this time, leading to increased structural stiffness [26].

To define a classification problem for the active learning experiments, the four natural frequencies are selected as the observation data, such that $\boldsymbol{x}_i \in \mathbb{R}^4$. Firstly, the damage data are assumed to represent their own class, from observation 3476. Outlying observations within the remaining dataset were then determined using the robust Minimum Covariance Determinant (MCD) algorithm [25,26]. These outlying data are illustrated in Fig. 7a; as discussed, they appear to relate to cold temperatures effects (specifically, observations 1200 and 1500). A 3-class classification problem can now be defined, such that $y_i \in \{1, 2, 3\}$:

- class 1: the normal data,
- class 2: outlying data due to environmental effects,
- class 3: damage.

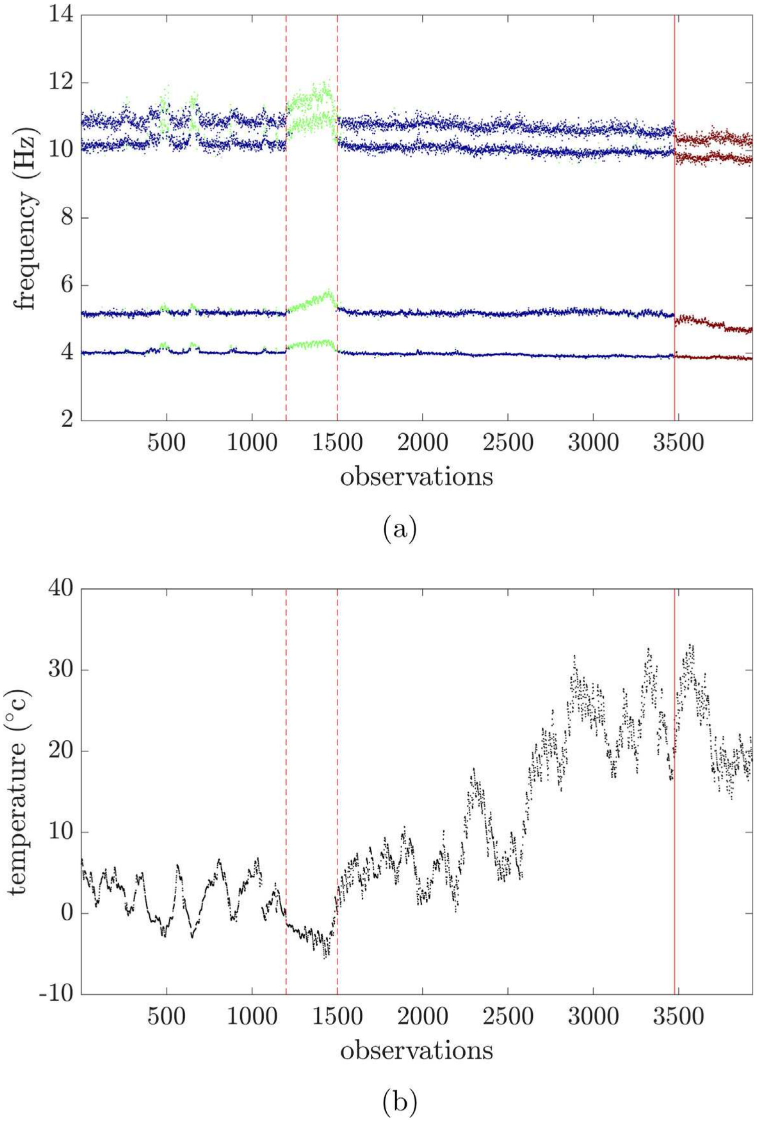


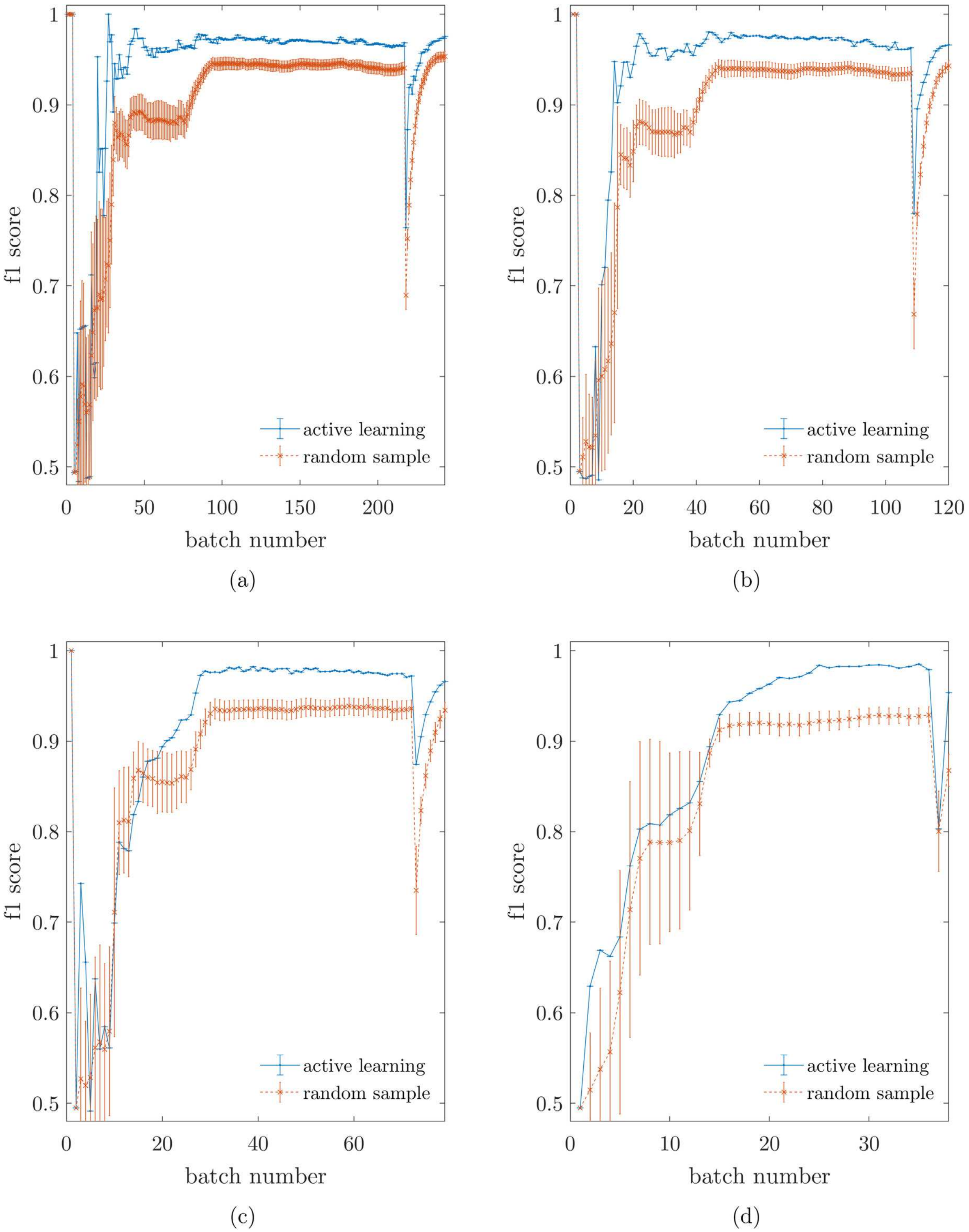
Fig. 7. Z24 bridge data: (a) time history of natural frequencies, (b) time history of average deck temperature.

In this application, it is clearly undesirable for an engineer to investigate the structure following each data acquisition from the bridge. Therefore, if active learning can provide an improved classification performance, compared to passive learning (random sampling) with the same sample budget, this demonstrates the relevance of active methods.

5.1.1. Results

Plots are provided for an increasing label budget per batch of data. As discussed, the dataset is split in half, to define the training set and test set; i.e. each subset contains 1966 observations for the Z24 data. Both sets increase at the same rate, such that measured data arrive as if recorded from the system in operation. The f_1 score is assessed using the test set. The queries per batch are kept constant with $q_b = 2$, while the batch size is increased, such that $B \in \{8, 16, 24, 48\}$. These values correspond to query ratios of 1:4, 1:8, 1:12 and 1:24, for labelled to unlabelled data respectively. Active learning (uncertainty sampling) and the passive learning benchmark (random sampling) are applied 50 times for each query-budget ratio. The results are provided in Fig. 8, error bars illustrate the one-sigma (σ) deviation.

Active learning for guided sampling successfully directs queries for an increased classification performance with these data. For all query budgets, there is a clear increase in the f_1-score when uncertainty sampling is used to build the training-set, $\mathcal{D}_l$. At times, sampling bias appears to negatively effect the f_1-score metric; specifically, in the early stages of monitoring, when 1:12 data are queried in Fig. 8c. In general, however, the increase in the classification performance appears



(a)

(b)

(c)

(d)

Fig. 8. Online classification performance (f_1-score) for the Z24 data, for query budgets (as ratios): (a) 1:4; (b) 1:8; (c) 1:12, (d) 1:24.

to outweigh the risk for this application. As expected, there are drops in the classification performance as new classes are discovered by the learner; however, these are less exaggerated when an active framework is used. (The drops in performance occur as the macro-averaged f_1 score weights each class equally.)

Another advantage for active learning is consistent model predictions; this occurs because data selection follows a deterministic process. In other words, the active learner will always select the same observations, if identical data are presented in the same order. As a result, the f_1-scores are consistent, because the variability associated with the 'informativeness' of a random sample is eliminated. For lower query budgets (Fig. 8c and d) while active learning increases the performance, the classifier does not have enough information to build a reliable model of the data; thus the f_1 scores are particularly low for both active and passive learning. To combat this issue, the query regime must be adapted (to catch the novel classes sooner), or the model should be updated to deal with this lack of information; these ideas are discussed in the conclusions.

5.2. Machining data

The machining data are an acoustic emission dataset, collected by Wickramarachchi et al., during experiments concerning a turning operation, used to manufacture metallic components [27]. During normal operation, the cutting tool deteriorates, leading to tool wear, see Fig. 9. Tool wear is undesirable, as it produces a poor surface finish for the machined component, which can lead to the onset of crack propagation, reducing the time in service for the manufactured product [28]. Consequently, it is critical to monitor wear of the tool; however, the current procedure requires the machining operation to be stopped, to allow for manual inspection. As a result, these inspections are infeasible in practice, due to cost and time implications [27], thus, the high-value cutting tools may be discarded prematurely when used in industry. For the experimental dataset used in this work, inspection of the tool is carried out using a 3D microscope, the resulting images are illustrated in Fig. 9.

Significant cost savings can be achieved if a model is capable of tool wear predictions while using a minimal number of tool inspections. In order to build a model to predict the current state of wear, acoustic emission (AE) measurements were taken during a typical machining operation, until catastrophic failure of the tool — see Fig. 9b. Measurements were made by placing an AE sensor on the machine turret; these data were recorded in the time domain, and then converted into the frequency domain. Following various signal processing steps, the measured data have 129 dimensions, with 1729 observations. For further details, see [27] — in this work, the measured data were collected using a similar experimental procedure, however, these tests concern the collection of data for a different machining operation. The data are then compressed through a random projection; this method for dimension reduction is frequently used in the compressive sensing literature [29], and it is applied to online SHM in [9]. Using this approach, a random matrix is generated and used to project the data on to 20 dimensions in an online manner, as each new batch of data arrives. 20 features were chosen as this produced a relatively challenging feature space for the classification problem. Therefore, the measured data are defined such that $\boldsymbol{x}_i \in \mathbb{R}^{20}$. As the annotation of these measurements is expensive, the tool was inspected at 10 regular intervals during the experiments. This corresponds to 9 different classes (ranges) of tool wear, and one class after tool failure, such that $y_i \in \{1, \ldots, 10\}$. Table 1 summarises the dataset as a classification problem.

By using AE measurements, such as the dataset presented in this work, it is desirable to accurately monitor tool wear online, while keeping the number of tool investigations (to annotate the measured data) to a minimum. Considering this aim, the active learner is applied to the machining data sequentially, as if it were online. As with all the experiments, the

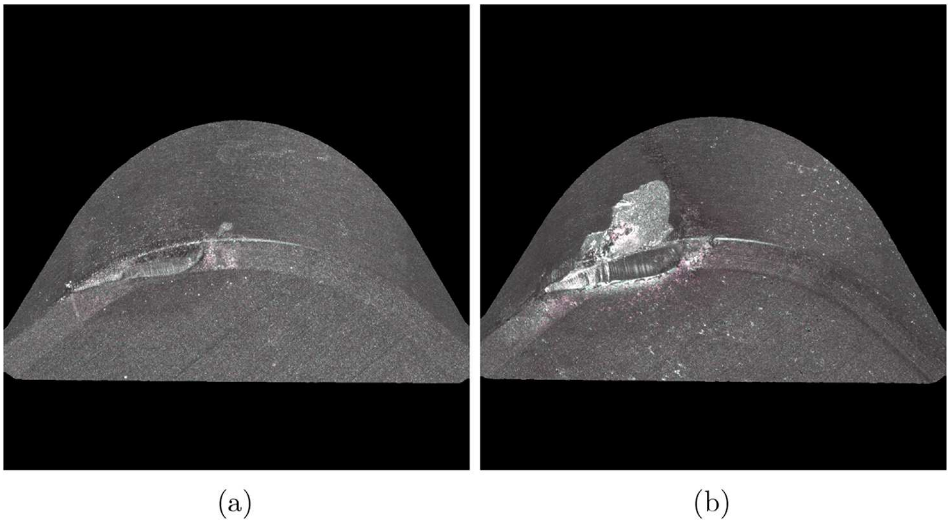


Fig. 9. Tool wear following inspection: (a) minor tool wear, (b) catastrophic failure of the tool.

Table 1
Machining AE data classes.

Class label (y_i)	Observations (i)	Description
1	1–173	wear 1
2	174–346	wear 2
3	347–519	wear 3
4	520–692	wear 4
5	693–865	wear 5
6	866–1038	wear 6
7	1039–1211	wear 7
8	1212–1383	wear 8
9	1384–1555	wear 9
10	1556–1729	tool failure

class labels, y_i, are hidden from the algorithm, and only measurements queried by the learner are provided with labels. Therefore, this framework implies that the engineer only needs to investigate the system when the learner queries.

5.2.1. Results

In these tests, the batch size is increased such that $B \in \{8, 16, 24\}$. This corresponds to query ratios of 1:4, 1:8, and 1:12, for labelled to unlabelled data respectively. Again, the sample budget per batch is $q_b = 2$, and active/passive learning methods are applied 50 times. Plots are provided in Fig. 10. Active learning brings consistent improvements to the classification performance with the machining data, although, these advantages are less significant — note the reduced axis range for the f_1 score. It is believed this occurs because the data are relatively separable in the feature space, thus, the use of active learning is less effective. Intuitively, a multi-class classification problem that is less mixed in the feature space should benefit less from active learning. Nevertheless, uncertainty sampling provides an increase in the classification performance at low query budgets; particularly when 1 in 12 data are labelled, see Fig. 10c. Fig. 10a shows that active learning can still be utilised at high query budgets for these data, as the variability of the prediction is reduced, such that the performance of active learner is comparable to the upper bound (1σ) of the expected performance for random sampling, see Fig. 10a and b. Furthermore, for all query budgets, the active learner appears to be more resilient to significant drops in the classification performance, particularly when new classes are introduced. This effect is most likely due to low-likelihood queries successfully targeting data relating to new classes, thus identifying them (and incorporating them into the model) sooner than random sampling. The variation in the classification performance across active learning is the result of the random projections, and not the active learning heuristic, which still builds the training-set deterministically. Likewise, the variation in the passive learning performance is also influenced by random projection, as well as random sampling.

5.3. Gnat aircraft data

The Gnat data are an experimental SHM dataset, recorded using a network of sensors placed on the wing of a Gnat aircraft; schematics are provided in Fig. 11. During experiments, the wing was excited using an electrodynamic shaker under white Gaussian excitation. Transmissibilities associated with nine selected inspection panels were used as the main measurements, see [30–32] for justification. The transmissibility is a frequency domain observation, and in this case, it is equivalent to the ratio of the Fourier transform of the response (transmitted) acceleration to that of the reference acceleration. The sensor layout is shown in Fig. 11b. The panels are split into three groups (A, B and C); each group has one centrally placed reference transducer and three response transducers; as such, the associated transmissibilities are also shown in Fig. 11b. In all cases, 1024 spectral lines were recorded, between 1024 and 2048 Hz [33]. The logarithms of the transmissibility magnitudes are used as the input measurements.

During the experiments, artificial damage and maintenance procedures were simulated by sequentially removing and replacing each of the nine inspections panels. It should be considered that the removal of each panel imitates a fairly large, significant fault. Each panel is held in place with number of screws, ranging from 8 to 26. These were replaced using an electric screwdriver with controllable torque, in an attempt to keep constant boundary conditions [32]. As a result, these data represent a 10-class problem; one class is associated with the normal condition (including repairs) and one class for each state of damage (nine in total). There are 2500 observations in the dataset; 700 one-shot measurements for the normal condition and 200 for each damage condition [33]. The data are ordered such that they represent the true sequence of experiments [32]; therefore, each set of damaged tests is followed by a normal condition test. This is done to simulate an online SHM environment, where damage is followed by 'maintenance' procedures (panel replacement), bringing the structure back to the normal operating condition. Table 2 summarises the ordered dataset.

The complete measured data have 9216 features in total (1024×9). In the original papers [32,33], these data were compressed to 9-dimensions using 9 Mahalanobis-squared-distance (MSD) novelty detectors [1], one learnt from each transmissibility. In the proposed SHM framework [32], the discordancy outputs (from each novelty detector) were initially used for *damage detection*; secondly, *damage location* was achieved using the discordancy measures as inputs to learn a classifier [32]. This strategy is successful, however, it considers the supervised problem, such that labelled data are used to inform feature

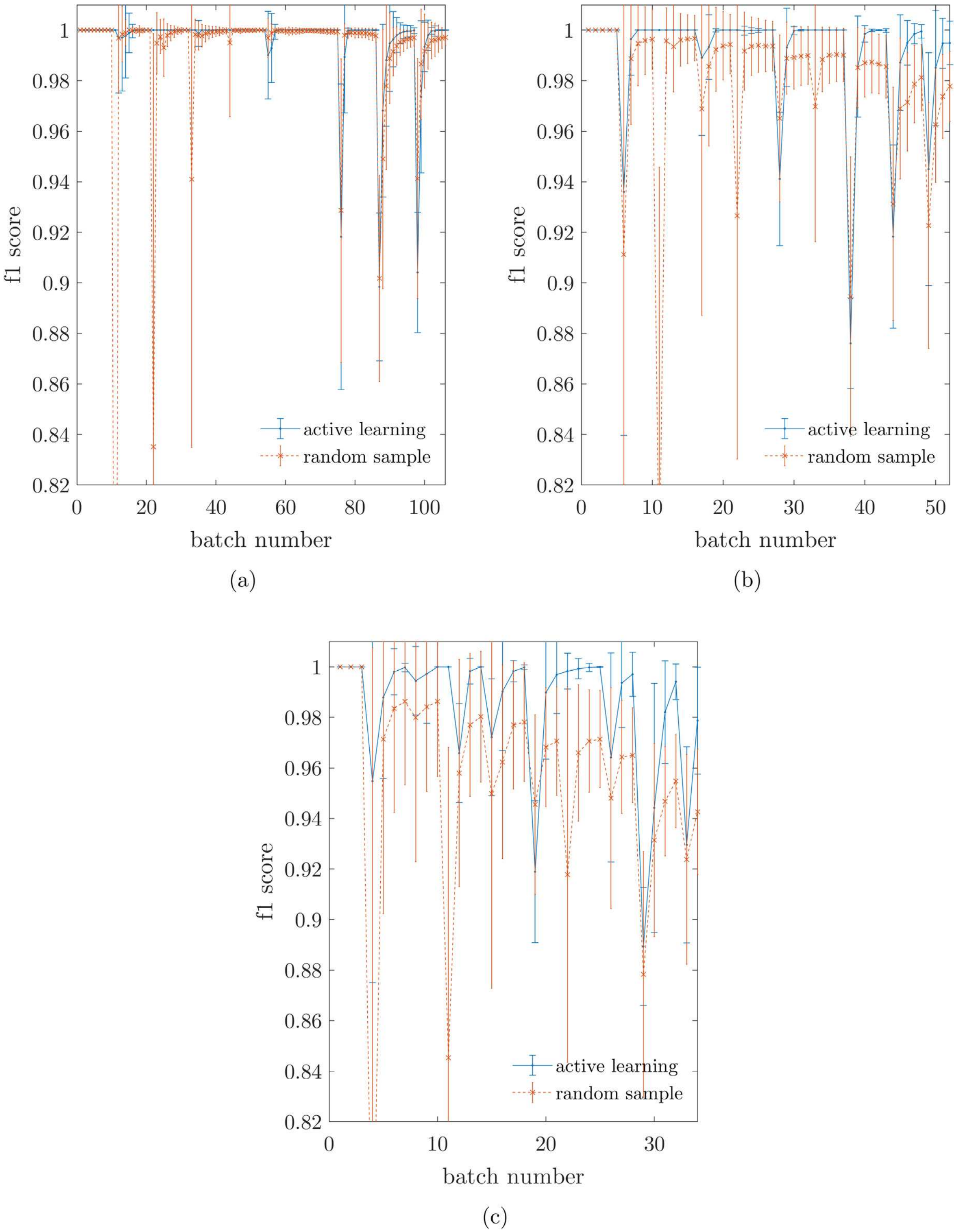


Fig. 10. Online classification performance (f_1-score) for the machining AE data, for query budgets (as ratios): (a) 1:4; (b) 1:8; (c) 1:12.

selection and dimension reduction. (This is either done objectively [32], or using a genetic algorithm to compress the data using a labelled training-set [33].) In the case of active learning, labelled data are (initially) unavailable; therefore, measurements must be compressed in an unsupervised manner. To do this, a feature bagging method is used to build a novelty index

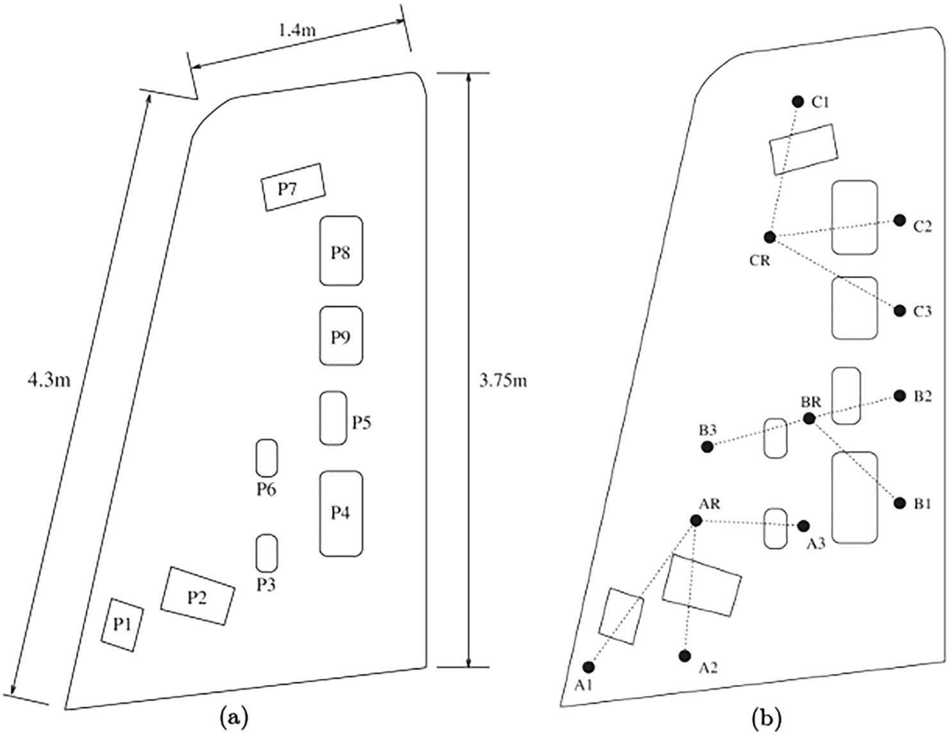


Fig. 11. Schematics of the Gnat aircraft wing: (a) panel locations, (b) sensor groups and transmissibilities. Image Credit: [32].

Table 2
Gnat data classes.

Class label (y_i)	Observations (i)	Description
1	1–100	normal
2	101–200	damage 1 (panel 1)
3	201–300	damage 2 (panel 2)
4	301–400	damage 3 (panel 2)
1	401–500	normal
2	501–600	damage 1 (panel 1)
3	601–700	damage 2 (panel 2)
4	701–800	damage 3 (panel 2)
1	801–900	normal
5	901–1000	damage 4 (panel 4)
6	1001–1100	damage 5 (panel 5)
7	1101–1200	damage 6 (panel 6)
1	1201–1300	normal
5	1301–1400	damage 4 (panel 4)
6	1401–1500	damage 5 (panel 5)
7	1501–1600	damage 6 (panel 6)
1	1601–1700	normal
8	1701–1800	damage 7 (panel 7)
9	1801–1900	damage 8 (panel 8)
10	1901–2000	damage 9 (panel 9)
1	2001–2100	normal
8	2101–2200	damage 7 (panel 7)
9	2201–2300	damage 8 (panel 8)
10	2301–2400	damage 9 (panel 9)
1	2401–2500	normal

for each tranmissibility. Details of how these features are defined in an unsupervised setting can be found in [34]. Briefly, MSD outlier ensembles are defined using random subsets of features (bootstrap samples). The novelty indices from each member in the ensemble are then combined through averaging to provide a single (robust) novelty index from high dimensional data. In this way, an outlier ensemble is built for each transmissibilty, compressing the dataset to nine dimensions in an unsupervised manner (such that only the normal condition data are used). In summary, this is now a 10-class classification problem in nine dimensions; one class defines the normal operating condition and 9 for the damaged states, where $y_i \in \{1, \ldots, 10\}$ and $\boldsymbol{x}_i \in \mathbb{R}^9$.

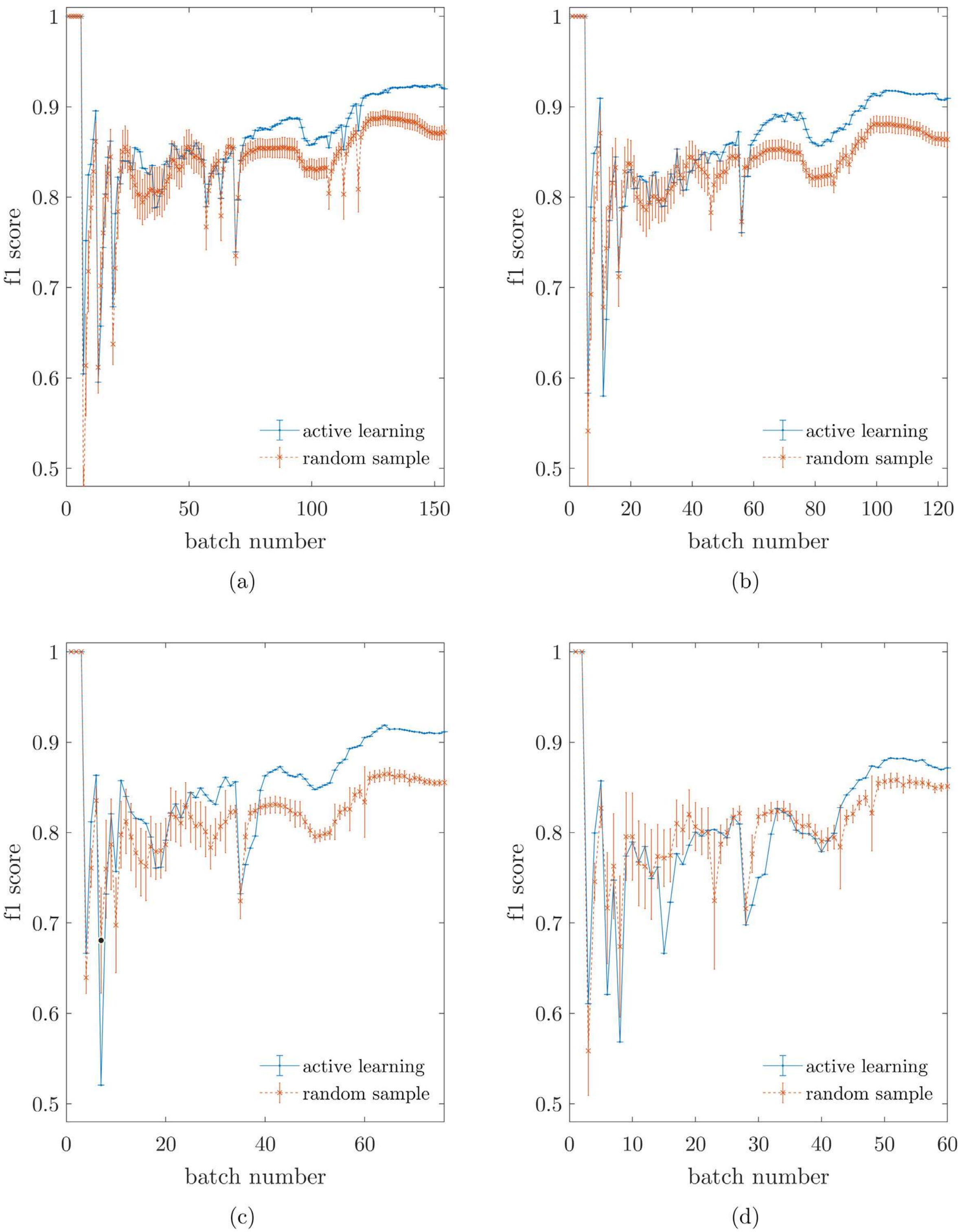


Fig. 12. Online classification performance (f_1-score) for the Gnat data, for query budgets (as ratios): (a) 1:4; (b) 1:5; (c) 1:8, (d) 1:10.

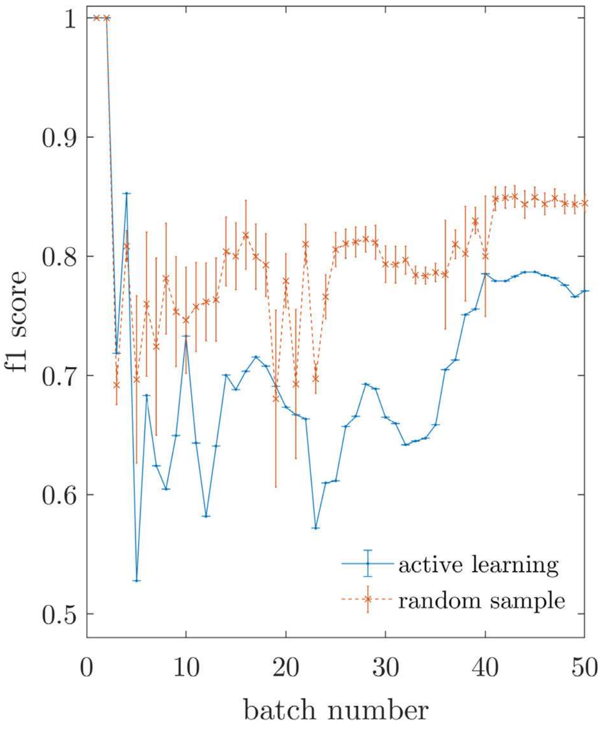


Fig. 13. Online classification performance (f_1-score) for the Gnat data, for a query budget of 1:12. The results show significant sampling bias, which is detrimental to the classification performance.

5.3.1. Results

For the Gnat data, the batch size is varied for $B \in \{8, 10, 16, 20, 24\}$ (while $q_b = 2$) to show a range of active learning effects. This corresponds to query ratios of 1:4, 1:5, 1:8, 1:10 and 1:12, for labelled to unlabelled data. As before, the results in Fig. 12 show improvements when uncertainty sampling is used; particularly for high query budgets, shown in Fig. 12a, b and c. With the Gnat data, the improvements appear to become less significant as the query budget decreases. This implies that active learning fails to provide significant improvements as the learner is allowed to query less. To investigate this further, the hueristic is run for a 1:12 query budget; the results are shown in Fig. 13, and demonstrate a clear example of sampling bias. In this case, the performance of active learning is worse than standard passive learning (random sampling); as discussed, this phenomenon is well established as a critical issue when applying active learning [11,19,21].

It is hypothesised that the performance of active learning deteriorates at low query budgets because the Gnat data represent a particularly difficult classification problem, with 10 classes in a mixed feature space. While the complexity of the classification means that active learning can bring significant advantages at high query budgets (Fig. 12a, b and c), once the number of queries passes a critical point ($\sim$1:10), the data become misrepresentative of the underlying distribution; in consequence, there is not enough information in the model to successfully direct queries in a way that benefits the classification. These results are important, as they imply that while active learning is useful for complex online classification, if the sample budget is too low, it can have a detrimental effect on the performance. As a result, is it critical that a method is defined to establish when (and how much) querying is required; this idea is being considered for future work.

6. Future work

While this model works well for these data, the fact that this is a *parametric-statistical* model should be considered; in other words, assumptions are made about the distribution of the measured data. If the classes of data form disjoint (multi-modal) clusters in the feature space, this active framework might still bring advantages compared to random sample training for the same classifier; however, it is unlikely that the performance of either method would compare to that of nonparametric classifiers. (Nonparametric refers to the method used to describe the data distribution.) Some examples of such algorithms include: Gaussian process classification, relevance vector machines, or support vector machines [4]. Importantly, it is desirable to build an active learner around probabilistic measures in engineering (as in this work) as these models provide uncertainties with the associated predictions; however, a more general framework might be achieved by using a nonparametric approach, which does not make assumptions regarding the distribution of the data in X — such as the framework suggested in [9].

Most critically, a method must be defined to determine when and how much data to query in the online setting for active learning in SHM. In this work, a fixed number of measurements were queried with each batch of data; however, the algorithm might perform better if data are sampled only when necessary. In this way, the algorithm could choose when and which data to query, based on properties of the probabilistic model. Additionally, the automation of when to query should protect against too few data being sampled, which has been shown to lead to sampling bias with the Gnat data. Finally, the sampling regime could determine which type of data to query (i.e. high entropy, low likelihood, or another measure), providing further automation to the SHM strategy.

7. Conclusions

The comprehensive annotation of engineering datasets is costly/infeasible due to practical limitations; therefore, active learning techniques are suggested, such that only the most informative observations are labelled. This paper defines a probabilistic approach to guide data queries in a novel strategy for online structural health monitoring. The model is initialised as a one-class classifier (novelty detection) and adapts online as new classes are discovered — becoming a probabilistic multi-class classifier. In the experiments, the heuristic is applied to three datasets: the Z24 bridge data, a machining (acoustic emission) dataset, and a vibration-based dataset from a Gnat aircraft. The active learning algorithm is applied to the measurements as if they were online, recorded live from the systems in operation. Generally, the results show a clear increase in the online diagnostic performance of the probabilistic classifier, when active learning is used to build the training-set through uncertainty sampling; this is compared to standard passive learning, where the same number of observations are investigated at random. Furthermore, the variability of the classification performance is significantly reduced when active learning is utilised. It is important to note that there are issues concerning sampling bias at low query budgets, particularly for the Gnat data. However, the definition of a probabilistic method to determine *when* to query (i.e. the optimal query budget) is currently being investigated for future work.

Acknowledgements

The authors gratefully acknowledge the support of the UK Engineering and Physical Sciences Research Council (EPSRC) through Grant references EP/R003645/1, EP/R004900/1, EP/I01800X/1 and EP/S001565/1. Further thanks are extended to Karen Holford and Rhys Pullin at Cardiff University for providing the AE data, and to Element Six Ltd. for granting permission to use the machining data.

References

[1] C.R. Farrar, K. Worden, Structural Health Monitoring: A Machine Learning Perspective, John Wiley & Sons, 2012.
[2] M.M. Moya, M.W. Koch, L.D. Hostetler, One-class classifier networks for target recognition applications, Proceedings World Congress on Neural Networks, 1993, pp. 359–367.
[3] F. Schwenker, E. Trentin, Pattern classification and clustering: a review of partially supervised learning approaches, Pattern Recogn. Lett. 37 (1) (2014) 4–14.
[4] K.P. Murphy, Machine Learning: A Probabilistic Perspective, MIT press, 2012.
[5] O. Chapelle, B. Scholkopf, A. Zien, Semi-Supervised Learning, MIT Press, 2006.
[6] L. Bull, K. Worden, G. Manson, N. Dervilis, Active learning for semi-supervised structural health monitoring, J. Sound Vib. 437 (2018) 373–388.
[7] M. Wang, F. Min, Z.H. Zhang, Y.X. Wu, Active learning through density clustering, Expert Syst. Appl. 85 (2017) 305–317.
[8] X. Zhu, P. Zhang, X. Lin, Y. Shi, Active learning from data streams, Seventh IEEE International Conference on Data Mining (ICDM 2007), 2007, pp. 757–762.
[9] T.J. Rogers, K. Worden, R. Fuentes, N. Dervilis, U.T. Tygesen, E.J. Cross, A Bayesian non-parametric clustering approach for semi-supervised Structural Health Monitoring, Mech. Syst. Signal Process. 119 (2019) 100–119.
[10] L. Bull, G. Manson, K. Worden, Dervilis, Active learning approaches to structural health monitoring, in: N. Dervilis (Ed.), Special Topics in Structural Dynamics, vol. 5, Springer International Publishing, 2019, pp. 157–159.
[11] S. Dasgupta, D. Hsu, Hierarchical sampling for active learning, in: Proceedings of the 25th International Conference on Machine Learning ACM, 2008, pp. 208–215.
[12] S. Chen, F. Cerda, P. Rizzo, J. Bielak, J.H. Garrett, J. Kovačević, Semi-supervised multiresolution classification using adaptive graph filtering with application to indirect bridge structural health monitoring, IEEE Trans. Signal Process. 62 (11) (2014) 2879–2893.
[13] A.K. McCallumzy, K. Nigamy, Employing EM and pool-based active learning for text classification, in: Proc. International Conference on Machine Learning (ICML), 1998, pp. 359–367, Citeseer.
[14] A. Gelman, H.S. Stern, J.B. Carlin, D.B. Dunson, A. Vehtari, D.B. Rubin, Bayesian Data Analysis, Chapman and Hall/CRC, 2013.
[15] B.C.M. Pattern, Recognition and Machine, Learning. Springer, 2006.
[16] D.J. MacKay, Information Theory, Inference and Learning Algorithms, Cambridge University Press, 2003.
[17] G. Manson, K. Worden, K. Holford, R. Pullin, Visualisation and dimension reduction of acoustic emission data for damage detection, J. Intell. Mater. Syst. Struct. 12 (2001) 529–536.
[18] S. Rippengill, K. Worden, K.M. Holford, R. Pullin, Automatic classification of acoustic emission patterns, Strain 39 (2003) 31–41.
[19] B. Settles, Active learning, Synthesis Lectures Artificial Intell. Mach. Learn. 6 (1) (2012) 1–114.
[20] S.J. Huang, R. Jin, Z.H. Zhou, Active learning by querying informative and representative examples, Advances in Neural Information Processing Systems, 2010, pp. 892–900.
[21] H.T. Nguyen, A. Smeulders, Active learning using pre-clustering, Proceedings of the Twenty-first International Conference on Machine learning, ACM, 2004, p. 79.
[22] M. Song, H. Wang, Highly efficient incremental estimation of Gaussian mixture models for online data stream clustering, Intelligent Computing: Theory and Applications III, vol. 5803, International Society for Optics and Photonics, 2005, pp. 174–184.
[23] G.D. Roeck, The state-of-the-art of damage detection by vibration monitoring: the SIMCES experience, Struct. Control Health Monit. 10 (2) (2003) 127–134.

[24] B. Peeters, G. De Roeck, One-year monitoring of the Z24-bridge: environmental effects versus damage events, Earthquake Eng. Struct. Dyn. 30 (2) (2001) 149–171.

[25] N. Dervilis, E. Cross, R. Barthorpe, K. Worden, Robust methods of inclusive outlier analysis for structural health monitoring, J. Sound Vib. 333 (20) (2014) 5181–5195.

[26] P.J. Rousseeuw, K.V. Driessen, A fast algorithm for the minimum covariance determinant estimator, Technometrics 41 (3) (1999) 212–223.

[27] C. Wickramarachchi, T. McLeay, S. Ayvar-Soberanis, W. Leahy, E. Cross, Tool wear inspection of polycrystalline cubic boron nitride inserts, Special Topics in Structural Dynamics, vol. 5, Springer, 2019, pp. 259–266.

[28] N. Ghosh, Y. Ravi, A. Patra, S. Mukhopadhyay, S. Paul, A. Mohanty, A. Chattopadhyay, Estimation of tool wear during CNC milling using neural network-based sensor fusion, Mech. Syst. Signal Process. 21 (1) (2007) 466–479.

[29] Y.C. Eldar, G. Kutyniok, Compressed Sensing: Theory and Applications, Cambridge University Press, 2012.

[30] K. Worden, G. Manson, D. Allman, Experimental validation of a structural health monitoring methodology: Part I. Novelty detection on a laboratory structure, J. Sound Vib. 259 (2) (2003) 323–343.

[31] G. Manson, K. Worden, D. Allman, Experimental validation of a structural health monitoring methodology: Part II. novelty detection on a gnat aircraft, J. Sound Vib. 259 (2) (2003) 345–363.

[32] G. Manson, K. Worden, D. Allman, Experimental validation of a structural health monitoring methodology: Part III. Damage location on an aircraft wing, J. Sound Vib. 259 (2) (2003) 365–385.

[33] K. Worden, G. Manson, G. Hilson, S. Pierce, Genetic optimisation of a neural damage locator, J. Sound Vib. 309 (3) (2008) 529–544.

[34] L. Bull, K. Worden, R. Fuentes, G. Manson, J. Cross, and N. Dervilis. Outlier ensembles: Robust methods for damage detection and unsupervised feature extraction from high dimensional data. Manuscript submitted for publication (JSV)..