Article:
Bull, L., Worden, K. and Dervilis, N. orcid.org/0000-0002-5712-7323 (2020) Towards semi-supervised and probabilistic classification in structural health monitoring. Mechanical Systems and Signal Processing, 140. ISSN 0888-3270

https://doi.org/10.1016/j.ymssp.2020.106653

Towards Semi-Supervised and Probabilistic Classification in Structural Health Monitoring

L. A. Bull*, K. Worden, N. Dervilis

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

Abstract

In practical applications of data-driven Structural Health Monitoring (SHM), recording labels for each of the measured signals can be infeasible and expensive. In consequence, conventional methods for (supervised) machine learning can become irrelevant in certain applications of damage classification. Semi-supervised methods, however, allow algorithms to learn from information in the available unlabelled measurements as well a limited set of labelled data. As such, this paper suggests a semi-supervised Gaussian mixture model for probabilistic damage-classification, informed by *both* labelled and unlabelled signals. The generative statistical model is shown to improve the classification performance, compared to supervised learning, with simulated and experimental SHM data, while requiring no further inspections of the system. Specifically, semi-supervised learning leads to 3.87% and 3.83% reductions in the classification error for the simulated and experimental datasets respectively. These results indicate that, through semi-supervised learning in SHM, the cost associated with labelling data could be managed, as the information in a small set of labelled signals can be combined with larger sets of unlabelled data.

Keywords: semi-supervised learning; damage classification; statistical modelling; signal processing; pattern recognition; structural health monitoring.

Email address: l.a.bull@sheffield.ac.uk (L. A. Bull*)

1. Introduction

In the data-driven approach to Structural Health Monitoring (SHM) [1], pattern recognition (i.e machine learning [2–4]) algorithms are used to inform the detection and classification of damage. Generally, this problem requires the classification of measured data into groups – corresponding to condition states of the monitored system. While there may be an abundance of measured data, descriptive labels for every recorded observation are often unavailable. These labels are critical, as they define the current operating, environmental, or damage condition. In almost all applications, however, labelling each observation becomes impracticable, as this information requires an engineer to inspect the system, often manually; this can be expensive, infeasible, and potentially impossible [1]. The absence of labels is significant in engineering applications of machine learning, as labelled data are required to learn with conventional supervised algorithms [2]. In consequence, most practical SHM strategies rely on unsupervised techniques (or one-class classifiers [5]); these methods enable damage detection, but do not allow for the classification of multiple data groups, which can inform SHM beyond novelty detection.

An alternative set of techniques, referred to as partially-supervised [6], offer another approach; specifically, the algorithms can simultaneously utilise measurements with and without descriptive labels. Semi-supervised learning, a subset of the partially-supervised methods, is applied in this work. Semi-supervised learning allows for information in a subset of labelled data to be used in conjunction with any unlabelled data.

A probabilistic and semi-supervised algorithm is proposed for multi-class classification in SHM, through a generative mixture model. The classifier is applied to simulated and experimental SHM data for demonstration, where incorporating information in the unlabelled data is shown to increase the diagnostic performance. In other words, the information in a set of unlabelled measurements can improve predictive performance of a classifier, *alongside* the available labelled data.

The layout of the paper is as follows. Section 2 provides an overview of conventional pattern recognition and introduces semi-supervised learning for SHM. Section 3 introduces Gaussian Mixture Models for damage classification, including semi-supervised updates (via. Expectation Maximisation) within

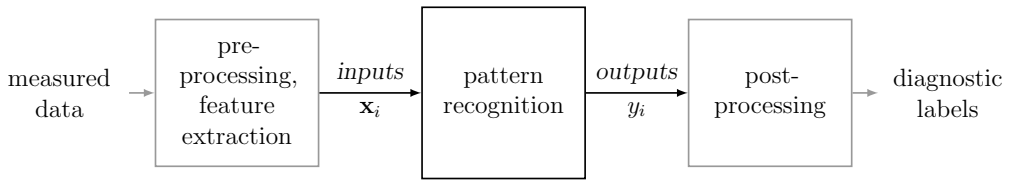


Figure 1: A framework for pattern recognition within SHM.

the probabilistic framework. Section 4 presents application of the algorithm to simulated and experimental data. Section 5 offers concluding remarks and future work.

2. Semi-supervised Learning for SHM

SHM strategies involve monitoring an engineering structure or system using observed sensor data to make informed predictions about the current (and future) condition of that system. In consequence, SHM can be viewed as a multi-class classification problem, such that measurements are categorised according to the correct operational, environmental, or damaged condition. Generally speaking, the i^{th} measured data point (or observation), $\mathbf{x}_i \in X$, is categorised according to a descriptive label, $y_i \in Y$, which corresponds to the ground truth of the classification problem. For SHM, each (potentially multivariate) observation, $\mathbf{x}_i$, represents features extracted from the raw measurements following pre-processing, while the descriptive labels, y_i, are used to specify the condition of the system, directly or indirectly; if indirectly, diagnostic labels can be inferred through some post-processing of the pattern recognition outputs y_i. The steps within a typical SHM strategy are shown in Figure 1.

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

2.1. Conventional machine learning for SHM

Supervised learning algorithms require a fully-labelled dataset for training, such that,

$$\mathcal{D}_l = \{(\mathbf{x}_i, y_i)\}_{i=1}^{n} \tag{1}$$

The training-set $\mathcal{D}_l$ includes both observation data and descriptive labels for n measured signals. As such, 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 inform diagnostic decisions in an SHM context.

Unsupervised learning, however, is applied when labels are unavailable. In this case, the training-set is [6],

$$\mathcal{D}_u = \{\tilde{\mathbf{x}}_i\}_{i=1}^{m} \tag{2}$$

with m observations. $\tilde{\mathbf{x}}_i$ is used herein to denote the measured data that are unlabelled. Various data analysis and machine learning tools can be applied to unlabelled datasets. Some examples of methods include: dimensionality reduction, novelty detection or outlier analysis, and clustering [2]. These techniques aim to find patterns within a dataset from the information contained within the measured observations alone; therefore, 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 [6].

As discussed, fully-labelled datasets are rarely feasible in practical SHM. Currently, this fact forces a dependence on conventional unsupervised techniques, such as novelty detection or one-class classifiers [5]. When working with engineering data, however, it is often possible to include labels for a small subset of measurements [7]. In this case, it is illogical to apply supervised learning to the small subset of labelled data, while ignoring information in a (potentially large) set of unlabelled data. Similarly, it is unjustified to ignore the available labels, in order to apply unsupervised algorithms. In this scenario, *partially-supervised* methods become relevant to SHM.

2.2. Partially-supervised learning

Partially-supervised algorithms [6] make use of both labelled data and unlabelled data, such that the training-set becomes,

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

where $\mathcal{D}_l$ are labelled data, and $\mathcal{D}_u$ are unlabelled data. Two of the main partially-supervised methods are *semi-supervised* and *active* learning. Active algorithms query and annotate the unlabelled data in $\mathcal{D}_u$ to automatically extend the labelled set $\mathcal{D}_l$, such that the resultant increase in the classification performance is maximised. As such, only the most informative observations are queried, to make the most out of a limited labelling budget. Active learning has been applied to SHM data in the past, in the offline [7] and online setting [8]. The focus of this work, however, considers *semi-supervised* variants of partially-supervised learning.

Semi-supervised learning

Semi-supervised learning utilises both the labelled and unlabelled data to inform the classification mapping, $f : X \mapsto Y$. Typically, a semi-supervised learner will use information in $\mathcal{D}_u$ to further update the classifier learnt from $\mathcal{D}_l$. Unlabelled data can be incorporated in various ways. The most simple approach, *self-labelling* [7, 9], trains a classifier using $\mathcal{D}_l$, and then predicts the labels for the unlabelled set $\tilde{\mathbf{x}}_i$. The classifier is then retrained using the labelled and unlabelled data. In the new training-set, some labels in $\mathcal{D}$ are the ground truth, from the supervised data, and the others are *pseudo-labels*, predicted by the classifier. Self-labelling is simple and can be applied to any supervised algorithm; however, the effectiveness is highly dependent on the method of implementation, and the supervised algorithm within it [9].

A more defined perspective considers *low-density-separation* [9]; this assumption implies that the decision-boundary of a classifier lies in low density regions of the feature-space; as such, the distances between the decision-boundary and its closest points in X are maximised. The use of a maximum-margin algorithm, such as the Support Vector Machine (SVM) [10], is most common in this setting; for example, the Transductive SVM (TSVM) [11] uses both the labelled data and the unlabelled data to maximise the margin of the classifier – through iterative self-labelling steps.

Generative mixture models provide an alternative framework to incorporate unlabelled data [12, 13]. Specifically, generative models utilise the cluster assumption: 'if points are in the same cluster, they are likely to be of the same class' [9]. (Note, this does not necessarily imply that each class is represented by a single, compact cluster in the feature-space; instead, it implies that observations from different classes are unlikely to appear in the same cluster [9].) When following this approach to density estimation [4], a mixture of base-distributions are used to estimate the underlying distribution of the data, defined by $p(\mathbf{x}_i, y_i)$. Generative models can naturally account for labelled and unlabelled data, as the Expectation Maximisation (EM) algorithm (used to learn mixture models in the unsupervised case [2, 14]) can be modified to incorporate labelled data [12, 15]. A strength of generative methods is that knowledge of the data structure can be incorporated by modelling it – this is often available *a priori* in engineering applications. However, if the assumptions of the generative model prove to be unreasonable (e.g unsuitable base-distributions), the structure imposed by the model can decrease the predictive accuracy.

More recent developments in the literature include *graph-based* learners [16]; this involves building a graph, where the nodes represent observed data (labelled and unlabelled), and the edges represent the similarities between observations [17]. The *manifold* assumption is relevant here: 'the (high-dimensional) data lie (roughly) on a low-dimensional manifold' [9]. Conveniently, the manifold assumption addresses the curse-of-dimensionality [4], which leads to an increasingly sparse feature-space in high dimensions; in this setting, statistical learning and density estimation (through generative mixture models) becomes problematic. Generally, graph-based methods inform semi-supervised learning through the smoothness assumption (for supervised learning), applied to the manifold: if two observations are close in a high-density region, they are likely to share the same label [9]. In view of this, the graph structure can be used to propagate labels from the labelled signals to the unlabelled instances.

2.3. Applications to SHM

Semi-supervised methods can bring significant advantages to SHM. For example, consider a wind turbine in an offshore windfarm. It is only possible

to provide labels describing the condition of various components (such as the turbine blades) following manual inspection; this involves travelling to a remote offshore location, which is a high-cost procedure. By utilising semi-supervised tools, the cost associated with labelling data can be managed, as the information in a small set of labelled data can be combined with larger sets of unlabelled data, recorded from the monitored system.

Related work

Semi-supervised learning has been applied to SHM in previous work. In the context of bridge monitoring, Chen *et al.* introduce a graph-based approach for label propagation [17, 18]. Specifically, the objective-function of a multi-resolution classifier [19, 20] is modified, such that the weighting parameters are optimised over the labelled and the unlabelled data; additionally the graph-based classifier [17] within the heuristic is semi-supervised. The Shannon entropy [21] is used to approximate an uncertainty associated with the confidence vector over the predicted labels for the unlabelled data; this information is included in the cost function, which learns the weights of the multi-resolution classifier, as well as the filter-coefficients within each graph-based classifier [17].

Further work concerns the application of K-means [22] and fuzzy-C-means [23] for semi-supervised SHM. (Fuzzy-C-means [24] is an adaptation of K-means clustering [2, 10], such that each signal can belong to more than one cluster, according to membership weights.) Firstly, Huang *et al.* [23] use fuzzy-C-means within an online SHM strategy; the proposed method becomes partially-supervised during a *label-matching step*, where the unsupervised clusters are compared to known classes from the supervised data. Bouzenad *et al.* [22] define a similar online heuristic using K-means; in this case, new clusters are created when a distance-based threshold is broken within the unsupervised algorithm. These heuristics can be considered as *clustering with constraints* [9]; an alternative view of semi-supervised learning, where partial-supervision is introduced through constraints on an *unsupervised* algorithm.

Contribution

This work suggests an alternative perspective, through *generative-mixture-models* for probabilistic and semi-supervised damage classification. Provided certain assumptions hold, generative methods allow for predictions with well-defined uncertainty, under Kolmogoroff's axioms [25] – a significant advantage in risk-based applications[1]. Additionally, in an engineering context, prior knowledge of the structure of the measured data is often available (e.g. drifting data streams or uni-modal clusters in the feature-space). As discussed, this *a priori* knowledge is easy to include within a generative framework, through the model definition.

3. Probabilistic Mixture Models for Semi-supervised SHM

For engineering datasets, *assuming* a parametric-statistical model (for density estimation) can be justified, given prior knowledge of the application. For example, SHM data recorded from a mechanical system or structure should remain relatively consistent for a given operating, environmental, or health condition – synonymous with the consistent underlying physics[2] [1]. As such, in this work, each class associated with the monitored system is assumed to define a unimodal (single and compact) cluster in the feature space, X.

Specifically, the data are assumed to be generated by a Gaussian Mixture Model (GMM) [2, 4]. Therefore, the underlying distribution of the measured data $\mathbf{x}_i \in X$, for each class k, is described by a Gaussian distribution,

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

where k is used to index the class group, such that $k \in \{1, ..., K\}$; therefore, $\boldsymbol{\mu}_k$ is the mean and $\boldsymbol{\Sigma}_k$ is the covariance of the data $\mathbf{x}_i$ with label k (i.e. there are K Gaussian base-distributions). If the Gaussian distribution proves too restrictive in describing the data in each component (e.g. the class clusters

[1] For example, consider a *certain* prediction, which states an oil-rig is safe to use; this differs significantly to an *uncertain* prediction, leading to the same statement.

[2] In turn, this justifies the cluster-assumption for semi-supervised mixture-models.

are multi-modal), an alternative base-distribution should be selected. The examples in this work, however, are appropriately described by a GMM.

The discrete random variable, $y_i \in \{1, ..., K\}$, which describes the labels is assumed to be categorically distributed [3],

$$p\left(y_i\right) = \mathrm{Cat}(y_i \,|\, \boldsymbol{\lambda}) \tag{5}$$

$\boldsymbol{\lambda}$ is vector of *mixing proportions*, which is a histogram over the label values, such that $\boldsymbol{\lambda} = \{\lambda_1, ..., \lambda_K\}$ and $p\left(y_i = k\right) = \lambda_k$. Bayes' rule can be applied using (4) and (5) to define a generative classifier, used to predict the class associated with an unseen signal, $\mathbf{x}_i^*$ [2],

$$p(y_i^* = k \,|\, \mathbf{x}_i^*,\ \boldsymbol{\theta}) = \frac{p\left(\mathbf{x}_i^* \,|\, y_i^* = k,\ \boldsymbol{\theta}\right) p\left(y_i^* = k \,|\, \boldsymbol{\theta}\right)}{p(\mathbf{x}_i^* \,|\, \boldsymbol{\theta})} \tag{6a}$$

$$\boldsymbol{\theta} \triangleq \{\boldsymbol{\Sigma}, \boldsymbol{\mu}, \boldsymbol{\lambda}\} \tag{6b}$$

$$p(\mathbf{x}_i^* \,|\, \boldsymbol{\theta}) \triangleq \sum_{k=1}^{K} p\left(\mathbf{x}_i^* \,|\, y_i^* = k,\ \boldsymbol{\theta}\right) p\left(y_i^* = k \,|\, \boldsymbol{\theta}\right) \tag{6c}$$

To learn the model by semi-supervised learning, the parameter-set $\boldsymbol{\theta}$ is learnt using both labelled data $\mathcal{D}_l$ and unlabelled data $\mathcal{D}_u$.

3.1. Supervised Gaussian Mixture Models

The first step in the semi-supervised GMM follows conventional supervised-learning [2, 4]. In this work, Bayesian estimates of $\boldsymbol{\theta}$ are defined by treating each parameter as a random variable, and placing prior distributions over the possible outcomes. Bayesian estimates of $\boldsymbol{\theta}$ exhibit a number of desirable properties for this application: the model becomes self-regularising, to prevent overtraining and aid generalisation for accurate predictions given new data; additionally, *a priori* information about the structure of the data can be included, and the zero-count problem [2] or black swan paradox [3], can be accommodated for.

Considering Gaussian-distributed observations in the feature-space X, and a categorical distribution over the label-space Y, a natural choice in the prior is a Normal-Inverse-Wishart (NIW) over $\boldsymbol{\Sigma}$ and $\boldsymbol{\mu}$, and a Dirichlet (Dir)

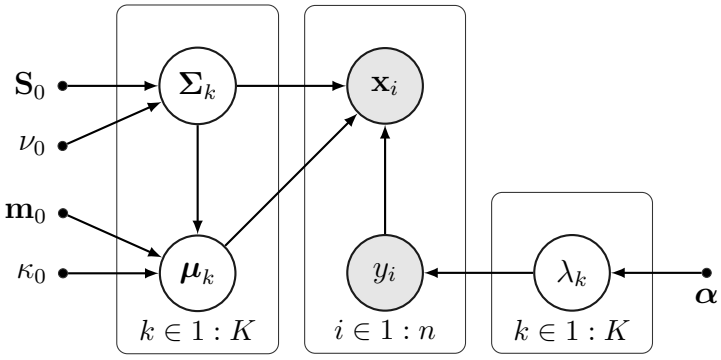


Figure 2: Graphical model for the GMM $p(\mathbf{x}_i, y_i, \boldsymbol{\theta})$ over the *labelled* data $\mathcal{D}_l$. As the dataset is supervised, both $\mathbf{x}_i$ and y_i are observed variables. (Shaded and white nodes are the observed and latent variables respectively; arrows represent conditional dependencies; dots represent constants (i.e. hyperparameters).)

distribution over $\boldsymbol{\lambda}$ [2, 3].

$$p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \text{NIW}\left(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k \,|\, \mathbf{m}_0, \kappa_0, \nu_0, \mathbf{S}_0\right) \tag{7}$$

$$p(\boldsymbol{\lambda}) = \text{Dir}(\boldsymbol{\lambda} \,|\, \boldsymbol{\alpha}) \tag{8}$$

$$\boldsymbol{\alpha} \triangleq \{\alpha_1, \ldots, \alpha_k\} \tag{9}$$

For the hyperparameters of the NIW, $\mathbf{m}_0$ is the prior mean for the location of $\boldsymbol{\mu}_k$, while κ_0 defines the strength of that prior; $\mathbf{S}_0$ is proportional to the prior mean of the covariance $\boldsymbol{\Sigma}_k$, while ν_0 defines the strength of that prior [2]. For the Dirichlet distribution, the hyperparameter $\boldsymbol{\alpha}$ incorporates prior belief in the mixing proportions for each class, λ_k. As such, α_k can be viewed as a vector of pseudo-counts, corresponding to the expected number of observations per class. These distributions are suitable, as they are conjugate to (4) and (5), leading to analytically-tractable solutions of the parameter estimates, defined in (10) and (11). A graphical model, *corresponding to the labelled data* $\mathcal{D}_u$, is shown in Figure 2.

In this application, the prior distributions encode the belief that the measured data are expected to be unit-variance and zero-mean (i.e. the feature-space is normalised), while each class in the mixture model is equally likely. In consequence, the hyperparameters are: $p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) = \text{NIW}(\mathbf{0}, 1, D, \mathbb{I})$, where $\mathbb{I}$ is a $[D \times D]$ identity matrix, and $\mathbf{0}$ is a D-dimensional vector of

zeros; and $p(\boldsymbol{\lambda}) = \mathrm{Dir}(\boldsymbol{\lambda} \,|\, \boldsymbol{\alpha})$, $\alpha_k = n/K$, $\forall k$. These prior assumptions are justified in this application, as it is possible to normalise the observation data in the feature-space X (both online and offline); furthermore, to represent a general case, if a class of data is observed infrequently in the labelled data, this does not (necessarily) imply that the class is less likely in the unlabelled data, or future measurements. However, if application specific knowledge is available, relating to the likelihood of a given class y_i, this information should be included via the Dirichlet prior. (For example, a group of data relating to abnormal temperatures might be associated with a low marginal probability $p(y_i)$ through the prior.)

Following these assumptions, the posterior distribution over the parameters $\boldsymbol{\mu}$ and $\boldsymbol{\Sigma}$, given the labelled data $\mathcal{D}_l$, is Normal-Inverse-Wishart [2, 3],

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

$$\mathbf{m}_n = \frac{\kappa_0}{\kappa_0 + n_k}\mathbf{m}_0 + \frac{n_k}{k_0 + n_k}\bar{\mathbf{x}}_k \tag{10b}$$

$$n_k \triangleq \sum_{i=1}^{n} \delta_{k,y_i} \tag{10c}$$

$$\bar{\mathbf{x}}_k \triangleq \frac{\sum_{i=1}^{n} \delta_{k,y_i}\, \mathbf{x}_i}{n_k} \tag{10d}$$

$$\kappa_n = k_0 + n_k \tag{10e}$$

$$\nu_n = \nu_0 + n_k \tag{10f}$$

$$\mathbf{S}_n = \mathbf{S}_0 + \mathbf{S}_k + \kappa_0 \mathbf{m}_0 \mathbf{m}_0^\top - \kappa_n \mathbf{m}_n \mathbf{m}_n^\top \tag{10g}$$

$$\mathbf{S}_k \triangleq \sum_{i=1}^{n} \delta_{k,y_i}\, \mathbf{x}_i \mathbf{x}_i^\top \tag{10h}$$

where δ_{k,y_i} is the Kronecker delta function, equal to 1 when k is equal to the observed class y_i, corresponding to observation $\mathbf{x}_i$. The bar notation $\bar{\mathbf{x}}_k$ is the empirical mean of the data in group k, and n_k is the number of observations in that group; finally, $\mathbf{S}_k$ is the uncentered sum-of-squares matrix for the data in class k (10h).

The Bayesian estimates of $\boldsymbol{\mu}_k$ (10b) and $\boldsymbol{\Sigma}_k$ (10g) are interpretable: the posterior mean $\mathbf{m}_n$ is a complex combination of the prior and the maximum-likelihood estimate; the posterior scatter matrix $\mathbf{S}_n$ is the prior scatter matrix,

plus the empirical scatter matrix, plus an additional term associated with uncertainty in the mean [2].

The posterior distribution over $\boldsymbol{\lambda}$ given the labelled data is [2],

$$p\left(\boldsymbol{\lambda} \,|\, \mathcal{D}_l\right) = \mathrm{Dir}(\boldsymbol{\lambda} \,|\, \{\alpha_1 + n_1, \ldots, \alpha_K + n_K\}) \tag{11}$$

Intuitively, the posterior is obtained by adding the pseudo-counts from the prior α_k to the empirical counts n_k.

The maximum *a posteriori* (MAP) estimate of the parameters $\hat{\boldsymbol{\theta}}$ corresponds to the mode of the posterior distribution,

$$\hat{\boldsymbol{\theta}} \mid \mathcal{D}_l \;=\; \{\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}, \hat{\boldsymbol{\lambda}}\} = \mathrm{argmax}_{\boldsymbol{\theta}} \left\{p(\boldsymbol{\theta} \,|\, \mathcal{D}_l)\right\} \;\therefore \tag{12a}$$

$$\hat{\boldsymbol{\mu}}_k = \mathbf{m}_n \tag{12b}$$

$$\hat{\boldsymbol{\Sigma}}_k = \frac{\mathbf{S}_n}{\nu_n + D + 2} \tag{12c}$$

$$\hat{\lambda}_k = \frac{\alpha_k + n_k - 1}{\sum_{k=1}^{K} \alpha_k + n - K} \tag{12d}$$

The posterior predictive equations are found by marginalising out the parameters from the model [2]; these equations are used to estimate (4) and (5) given the labelled data, $\mathcal{D}_l$, for the predictive classifier defined in (6),

$$\begin{aligned} p\left(\mathbf{x}_i^* \,|\, y_i^* = k, \mathcal{D}_l\right) &= \int\int p(\mathbf{x}_i^* \,|\, \boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k) p(\boldsymbol{\mu}_k, \boldsymbol{\Sigma}_k \,|\, y_i^* = k, \mathcal{D}_l) \; d\boldsymbol{\mu}_k d\boldsymbol{\Sigma}_k \\ &= \mathcal{T}\left(\mathbf{x}_i^* \,|\, \mathbf{m}_n, \; \frac{\kappa_n + 1}{\kappa_n \left(\nu_n - D + 1\right)} \mathbf{S}_n, \; \nu_n - D + 1\right) \end{aligned} \tag{13}$$

$$\begin{aligned} p\left(y_i^* = k \,|\, \mathcal{D}_l\right) &= \int p(y_i^* = k \,|\, \boldsymbol{\lambda}) p(\boldsymbol{\lambda} \,|\, \mathcal{D}_l) \; d\boldsymbol{\lambda} \\ &\propto \frac{\alpha_k + n_k}{\sum_{k=1}^{K} \alpha_k + n} \end{aligned} \tag{14}$$

where $\mathcal{T}$ is the Student-t distribution [2]. At this stage, the parameters that define equations (4) and (5) have been learnt using information in the labelled data only.

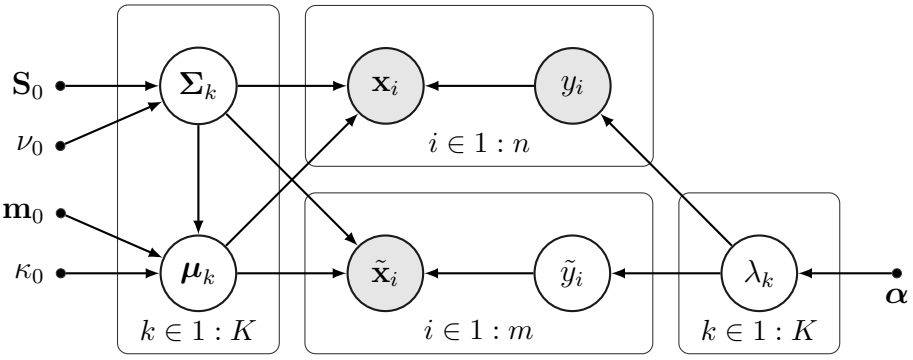


Figure 3: Graphical model of the GMM over both the *labelled* data $\mathcal{D}_l$ and the *unlabelled* data $\mathcal{D}_u$. For the unsupervised set, the only observed variable is $\tilde{\mathbf{x}}_i$, while $\tilde{y}_i$ is a latent variable.

3.2. Semi-supervised updates: Expectation Maximisation (EM)

The distribution over the parameters $\boldsymbol{\theta}$ is now updated using the unlabelled data $\mathcal{D}_u$. For the unlabelled observations, the label y_i can be considered a latent variable, herein denoted $\tilde{y}_i$. In this situation, the maximum *a posteriori* (MAP) estimate is more challenging to compute [2]. The Expectation Maximisation (EM) algorithm [14] is one method that solves this issue. The appropriate implementation of semi-supervised EM [15, 16] is similar to the unsupervised case, however, the log-likelihood of the model (and therefore the E/M-steps) are modified, such that the log-likelihood is maximised over both the labelled and the unlabelled data.

Specifically, the learning problem is defined to approach the MAP estimate of the parameters $\boldsymbol{\theta}$ given the labelled and unlabelled subsets, which is,

$$\begin{aligned} \hat{\boldsymbol{\theta}} \mid \mathcal{D} &= \operatorname{argmax}_{\boldsymbol{\theta}} \left\{ \frac{p(\mathcal{D} \mid \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathcal{D})} \right\} \\ &= \operatorname{argmax}_{\boldsymbol{\theta}} \left\{ \frac{p(\mathcal{D}_u \mid \boldsymbol{\theta}) p(\mathcal{D}_l \mid \boldsymbol{\theta}) p(\boldsymbol{\theta})}{p(\mathcal{D}_u, \mathcal{D}_l)} \right\} \qquad (15) \\ \mathcal{D} &\triangleq \mathcal{D}_u \cup \mathcal{D}_l \qquad (16) \end{aligned}$$

As such, it is assumed that $\mathcal{D}_u$ and $\mathcal{D}_l$ are conditionally independent. In this case, the assumption proves appropriate, as the training data are random samples from the underlying distribution: implicitly, random-sampling selects

representative data that are independent and identically distributed (i.i.d) [4]. For numerical stability, the MAP estimate is implemented as a maximisation of the expected joint log-likelihood of (15) across the complete dataset [9],

$$\begin{aligned}\mathcal{L}(\boldsymbol{\theta} \,|\, \mathcal{D}) &= \mathcal{L}(\boldsymbol{\theta} \,|\, \mathcal{D}_u, \mathcal{D}_l) \\ &\propto \sum_{i=1}^{m} \log \sum_{k=1}^{K} p\left(\tilde{\mathbf{x}}_i \,|\, \tilde{y}_i = k, \boldsymbol{\theta}\right) p(\tilde{y}_i = k \,|\, \boldsymbol{\theta}) \;\ldots \\ &+ \sum_{i=1}^{n} \log \left[p\left(\mathbf{x}_i \,|\, y_i = k, \boldsymbol{\theta}\right) p(y_i = k \,|\, \boldsymbol{\theta})\right] + \log p(\boldsymbol{\theta}) \end{aligned} \tag{17}$$

(The constant terms have been dropped for convenience.) As there exists a label y_i for each $x_i \in \mathcal{D}_l$, y_i is an observed variable for the term in (17) associated with the labelled data. However, in $\mathcal{D}_u$ the labels are unknown; therefore, the latent variable $\tilde{y}_i$ is marginalised out from the likelihood – this appears as a sum over k in (17). The model dependencies, including the observed and latent variables for each set, are illustrated in Figure 3.

In the EM algorithm, during each E-step, the *unlabelled* observations are classified using the current estimate of the model parameters and the classifier defined by (6). The M-step corresponds to finding the $\hat{\boldsymbol{\theta}}$ [3], given the *predicted labels* for unlabelled cases as well as the labelled data.

E-step. Initially, during the E-step, the responsibility matrix $\mathbf{r}$ is computed for the unlabelled data; this is the posterior distribution from the classifier defined in (6), thus, it is a $n \times K$ matrix,

$$r_{ik} = p(\tilde{y}_i = k \,|\, \tilde{\mathbf{x}}_i, \boldsymbol{\theta}) = \frac{p\left(\tilde{\mathbf{x}}_i \,|\, \tilde{y}_i = k, \boldsymbol{\theta}\right) p\left(\tilde{y}_i = k \,|\, \boldsymbol{\theta}\right)}{p(\tilde{\mathbf{x}}_i \,|\, \boldsymbol{\theta})}, \; \forall\, \tilde{\mathbf{x}}_i \in \mathcal{D}_u \tag{18}$$

The *effective counts* per class in $\mathcal{D}_u$ is the weighted number of points assigned to class k – this is the sum of the k^{th} column in the responsibility matrix, $r_k = \sum_{i=1}^{m} r_{ik}$ [2]. For the $\mathcal{D}_l$, however, the ground truth of $p(y_i = k \mid \mathbf{x}_i)$ is given by the training labels y_i; therefore, the posterior distribution is known

[3] Note, the initial estimate of $\hat{\boldsymbol{\theta}}$ is estimated from the labelled data only, and equations (10), (11), (12).

for the labelled points, which are discrete delta functions in the known class label [4],

$$p(y_i = k \,|\, \mathbf{x}_i) = \delta_{k,y_i}, \quad \forall\, (\mathbf{x}_i, y_i) \in \mathcal{D}_l \tag{19}$$

again, δ_{k,y_i} is the Kronecker delta function, which equals 1 when k is the observed label y_i. In summary, the total (effective) counts per class over the complete dataset are,

$$N_k = n_k + r_k \tag{20a}$$
$$N = |\mathcal{D}_l| + |\mathcal{D}_u| = n + m \tag{20b}$$

M-step. In each M-step, the equations used to update $\hat{\boldsymbol{\theta}}$ involve modifications to the supervised case, as defined in equations (10), (11), (12). Firstly, the vector of mixing proportions $\hat{\boldsymbol{\lambda}}$, for each element is,

$$\hat{\lambda}_k = \frac{\alpha_k + N_k - 1}{\sum_{k=1}^{K} \alpha_k + N - K} \tag{21}$$

The mean and covariance estimates are found by modifying (10), to give the parameters,

$$\mathbf{m}_n = \frac{\kappa_0}{\kappa_0 + N_k}\mathbf{m}_0 + \frac{N_k}{k_0 + N_k}\bar{\mathbf{x}}_k \tag{22a}$$
$$\bar{\mathbf{x}}_k \triangleq \frac{\sum_{i=1}^{n} \delta_{k,y_i}\, \mathbf{x}_i + \sum_{i=1}^{m} r_{ik}\tilde{\mathbf{x}}_i}{N_k} \tag{22b}$$
$$\kappa_n = k_0 + N_k \tag{22c}$$
$$\nu_n = \nu_0 + N_k \tag{22d}$$
$$\mathbf{S}_n = \mathbf{S}_0 + \mathbf{S}_k + \kappa_0 \mathbf{m}_0 \mathbf{m}_0^\top - \kappa_n \mathbf{m}_n \mathbf{m}_n^\top \tag{22e}$$
$$\mathbf{S}_k \triangleq \sum_{i=1}^{n} \delta_{k,y_i}\, \mathbf{x}_i \mathbf{x}_i^\top + \sum_{i=1}^{m} r_{ik}\, \tilde{\mathbf{x}}_i \tilde{\mathbf{x}}_i^\top \tag{22f}$$

Leading to the same equations for MAP estimation,

$$\hat{\boldsymbol{\mu}}_k = \mathbf{m}_n \tag{23a}$$
$$\hat{\boldsymbol{\Sigma}}_k = \frac{\mathbf{S}_n}{\nu_n + D + 2} \tag{23b}$$

The semi-supervised updates turn out to be simple and interpretable. The MAP estimates are similar to the supervised case in (10); however, information in $\mathcal{D}_u$ contributes to the counts (N and N_k), as well as the mean $\bar{\mathbf{x}}_k$ and scatter $\mathbf{S}_k$ estimates.

EM learning. The EM algorithm iterates between steps, leading to a hill-climbing search, which finds a *local* maximum in the parameter space. EM is sensitive to the initial estimate of $\hat{\boldsymbol{\theta}}$; to deal with this, the algorithm is normally initialised (randomly) many times. In this application, however, the starting point can be informed by the labelled data; as such, the initial guess is the MAP estimate given the labelled data, calculated with (10) and (11). This additional information mitigates the need to re-initialise the algorithm. Learning proceeds to iterate between E-steps (equations (18), (19)) and M-steps (equations (22), (23)), until the log-likelihood of the model, defined in (17), converges [14]. Semi-supervised EM is summarised in Algorithm 1.

Algorithm 1: *Semi-supervised EM for a Gaussian Mixture Model*

Input : Labelled data $\mathcal{D}_l$, unlabelled data $\mathcal{D}_u$
Output : Semi-supervised MAP estimates of $\hat{\boldsymbol{\theta}} = \left\{\hat{\boldsymbol{\mu}}, \hat{\boldsymbol{\Sigma}}\right\}$

Initilise $\hat{\boldsymbol{\theta}}$ using the labelled data, $\hat{\boldsymbol{\theta}} = \text{argmax}_{\boldsymbol{\theta}} \{p(\boldsymbol{\theta} \,|\, \mathcal{D}_l)\}$. Supervised GMM equations (10), (11) and (12);
while *the joint log-likelihood* $\mathcal{L}(\boldsymbol{\theta} \,|\, \mathcal{D})$ (17) *improves* **do**
    E-step: use the current model $p(\mathbf{x}_i, y_i, \hat{\boldsymbol{\theta}})$ to estimate class-membership for the unlabelled data $\mathcal{D}_u$ (18);
    M-step: update the MAP estimate of $\hat{\boldsymbol{\theta}}$ given the component membership for *all* observations $\hat{\boldsymbol{\theta}} := \text{argmax}_{\boldsymbol{\theta}} \{p(\boldsymbol{\theta} \,|\, \mathcal{D}_l \cup \mathcal{D}_u)\}$. Semi-supervised GMM equations (21), (22) and (23);
end

Following semi-supervised EM, the updated MAP estimates $\hat{\boldsymbol{\theta}}$ define the predictive classifier (6); this is used to predict the distribution over the class-labels for new observations $p(y_i^* \,|\, \mathbf{x}_i^*)$. Code for the semi-supervised GMM applied in this work is available via GitHub: `https://github.com/labull?tab=repositories`.

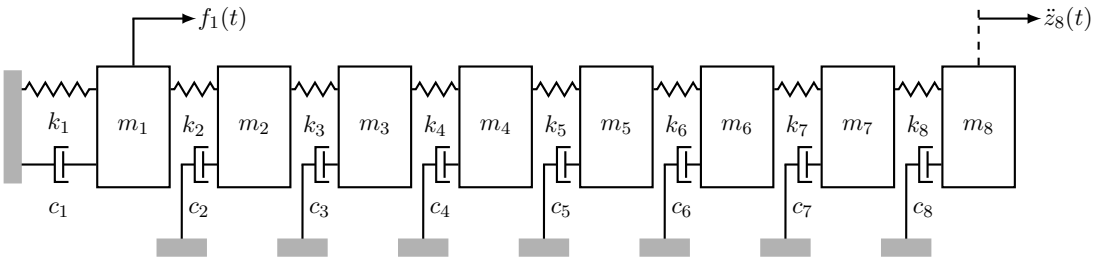


Figure 4: The simulated 8-DOF system

4. Experiments

Probabilistic and semi-supervised damage classification is applied to a simulated example and measured data from aircraft experiments. The simulated data demonstrate and visualise the model, while the experimental data present a more realistic and practical application.

4.1. Simulated Dataset

The simulated data represent measurements from an eight-degree-of-freedom (8-DOF) system. The system is defined to represent an experimental rig designed at the Los Alamos National Laboratory (LANL) [1]. A schematic of the 8-DOF system is shown in Figure 4[4]. The input forcing on mass i at time t is $f_i(t)$, and $z_i(t)$ is the system response (output) of mass i at time t.

The system parameters – mass m, stiffness k, damping c – are summarised in Table 1. The values for critical damping, c_c, are defined using the decoupled equations of motion. The system is set with approximately 3% of critical damping. The spring constant k_1 is set to near zero, as this corresponds to a rigid-body mode of the experimental rig. The forcing, $\mathrm{f}_1(t)$, is a white-noise excitation applied to mass 1, while the response, $\ddot{\mathbf{z}}(t)$, is simulated for all masses. Additive Gaussian noise is applied to the outputs, such that the signal-to-noise ratio (relative to variance) is 40dB.

It is expected that damage will manifest itself as alterations in the fundamental structural parameters; in this case, a reduction in stiffness [1]. Changes

[4]Note: there is repeated notation for the physical parameters m and k, however, the context and use of indices $(1-8)$ should make this clear.

Table 1: 8DOF system parameters

m_1	:	0.5993 kg
$\{m_2, ..., m_8\}$	:	0.4194 kg
k_1	:	10^{-6} kN/m
$\{k_2, ..., k_3\}$	:	56.7 kN/m
$\{c_1, ..., c_8\}$	:	$0.03 \times c_c$ Ns/m

in stiffness will alter the dynamic characteristics of the system; therefore, frequency domain observations can be used to (indirectly) monitor any physical changes that might relate to damage. In an attempt to represent SHM data, only the system outputs $\ddot{\mathbf{z}}(t)$ are used to define observations in the frequency domain. As such, the transmissibility between masses one and eight $T_{8,1}(\omega)$ is used a the frequency domain observation; this is a complex-valued function of frequency, which is the ratio of the spectrum of the output at mass eight, $\ddot{z}_8(t)$, to the spectrum of the output at mass one, $\ddot{z}_1(t)$. The transmissibility is approximated via the discrete Fourier transform of the output time series. A Hanning window is applied to each signal, sampled at 400.45Hz for 8 seconds. The transmissibilities are truncated, such that there are 1040 bins in the frequency domain, ranging from 0 - 130 Hz.

In terms of the SHM strategy, each transmissibility is an observation of the system; a transmissibility is generated every 8s from the time series data, and these data are used for monitoring. For demonstration, it is useful to compress the transmissibility data (1040-dimensions) onto two-dimensions using Principal Component Analysis (PCA) [10], to visualise the model[5]. The principal components are a linear combination of the original features, such that the variance is maximised in the projected space [2, 10]. As a result of PCA, observations $\mathbf{x}_i$ are two-dimensional, such that $\mathbf{x}_i \in \mathbb{R}^2$.

Linear damage is simulated as reductions in the spring constant k_5; the normal condition is when k_5 is at 100%, and a damage class is associated with each reduction in stiffness: there are five damage classes[6]. Generally, a

[5]The algorithm is applied to more realistic engineering data in the next experiment.

[6]Combinatorial damage could be considered for different spring locations; if included, these classes should have separate, distinct labels.

Table 2: Simulated data

Class label (y_i)	Observation index (i)	% k_5
1	1 - 500	100%
2	501 - 1000	97%
3	1001 - 1500	93%
4	1501 - 2000	88%
5	2001 - 2500	82%
6	2501 - 3000	70%

continuous parameter problem should not be framed as classification; however, discrete-steps are suitable to define a multi-class problem for this example. The data define a six-class problem, with 500 observations in each group; the data are summarised in Table 2, and the feature-space is shown in Figure 5.

Model visualisation: supervised learning vs. semi-supervised

The dataset is split (at random) into a training-set (2/3 of the total data, $\mathcal{D}$) and a test-set (1/3 of the total data, $\mathbf{x}_i^*$). 10% of the training-data $\mathcal{D}$ are labelled (the subset $\mathcal{D}_l$), while 90% remain unlabelled (the subset $\mathcal{D}_u$). The training subsets are shown in the feature-space in Figure 5.

Figure 5 plots the GMM for the supervised and semi-supervised case. In both plots, the prior is included to visualise it's influence on the base distributions of the mixture model. Specifically, with few data available for training, the prior should have a large influence on the posterior distributions, to regularise the model, as the parameters defined in (10b) and (10g) are a complex combination of the prior and the maximum-likelihood estimate.

Figure 5a shows the GMM given the labelled data only, i.e. $p(\mathbf{x}_i, y_i \,|\, \hat{\boldsymbol{\theta}})$ where $\hat{\boldsymbol{\theta}} = \text{argmax}_{\boldsymbol{\theta}} \{p(\boldsymbol{\theta} \,|\, \mathcal{D}_l)\}$. Here, the training data are a small subset, and, as a result, the prior has a large influence on base-distribution estimates. The influence of the prior is strong, as there is not enough information to appropriately model data, while avoiding overtraining. On the other hand, Figure 5b shows the mixture model can better represent the data distribution when unlabelled instances are used to inform the MAP estimates, such that $\hat{\boldsymbol{\theta}} = \text{argmax}_{\boldsymbol{\theta}} \{p(\boldsymbol{\theta} \,|\, \mathcal{D}_l, \mathcal{D}_u)\}$. Here, the base-distributions better represent each class, and the influence of the prior is reduced, while the model remains

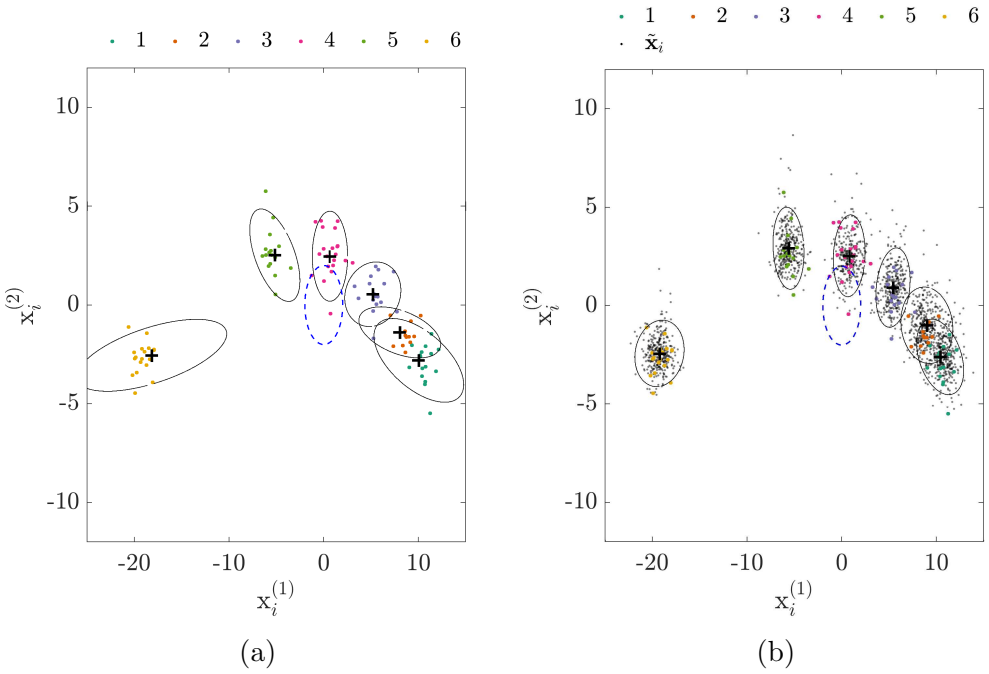


Figure 5: The GMM $p(\mathbf{x}_i, y_i \,|\, \hat{\boldsymbol{\theta}})$: (a) supervised learning, i.e. $\hat{\boldsymbol{\theta}} = \text{argmax}_{\boldsymbol{\theta}} \{p(\boldsymbol{\theta} \,|\, \mathcal{D}_l)\}$ (b) semi-supervised learning, i.e. $\hat{\boldsymbol{\theta}} = \text{argmax}_{\boldsymbol{\theta}} \{p(\boldsymbol{\theta} \,|\, \mathcal{D}_l, \mathcal{D}_u)\}$. Ellipses represent the MAP of the covariance (two-sigma), + markers represent the MAP of the mean, and the blue ellipse represents the prior.

self-regularised and robust.

It should be clear that the model is representative, as the density is well approximated by a GMM. If the data have multi-model class components, or the classes cannot (at least approximately) be represented by a Gaussian distribution, semi-supervised learning via a *Gaussian* mixture model will break down. In this case, an alternative base-distribution must be selected.

Classification test-procedure

The performance of the model (for classification) is assessed for an increasing number of labelled to unlabelled data. The proportion of labelled data in the training-set is increased in 5% increments, from 20% – 100%. For each proportion of labelled to unlabelled data, the GMM is initially learnt given

the labelled data only. Equation (6) is then used to classify the test-data, such that the predicted labels are the MAP of the posterior-distributions, $\hat{y}_i^* = \text{argmax}_k \{p(y_i^* = k \,|\, \mathbf{x}_i^*, \mathcal{D}_l)\}$. At this stage, the classification performance provides a benchmark for standard supervised learning.

The model is then updated via semi-supervised EM, given the labelled *and* unlabelled data. Label predictions are now the MAP estimates conditioned on the whole dataset, $\hat{y}_i^* = \text{argmax}_k \{p(y_i^* = k \,|\, \mathbf{x}_i^*, \mathcal{D}_l, \mathcal{D}_u)\}$. The classification performance is re-assessed for the semi-supervised model.

The metric used to assess classification performance is the f_1-score: this is a weighted balance of precision (P) and recall (R), which can be defined in terms of true positives (TP), false positives (FP) and false negatives (FN) for each class, $k \in Y$ [2],

$$P_k = \frac{TP_k}{TP_k + FP_k}, \quad R_k = \frac{TP_k}{TP_k + FN_k} \tag{24}$$

The macro f_1-score is then defined by [2],

$$\text{f}_{1,\text{k}} = \frac{2P_kR_k}{P_k + R_k}, \quad \text{f}_{1\text{macro}} = \frac{1}{K}\sum_{k \in Y} \text{f}_{1,\text{k}} \tag{25}$$

The macro-averaged f_1 is used, as this weights each class equally, mitigating any class-imbalance in the dataset. Therefore, newly-discovered groups in Y contribute equally to the performance metric, despite potentially infrequent observations; i.e. the novel measurements relating to damage or environmental conditions. For interpretability, in the context of SHM, the (balanced) misclassification error e (from type-I errors for each class) is also used as a performance metric,

$$\begin{aligned} e_k &= \frac{FP_k}{FP_k + TP_k} \\ e &= \frac{1}{K}\sum_{k \in Y} e_k \end{aligned} \tag{26}$$

Results

Figures 6 and 7 show the classification performance (f_1-score and error) for supervised and semi-supervised learning, while increasing the proportion

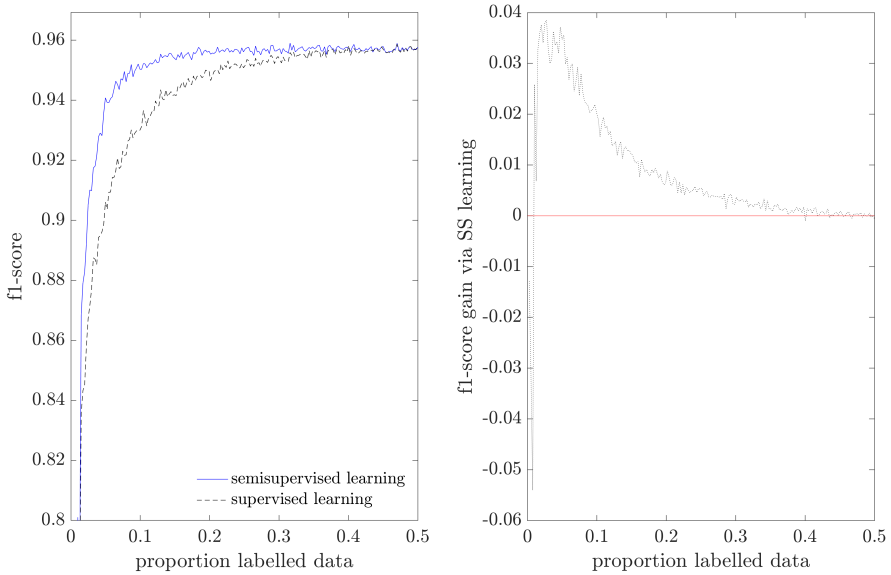


Figure 6: Classification performance assessed by the f_1-score for the supervised GMM vs. the semi-supervised GMM. Left: classification performance for an increasing proportion of labelled data. Right: the gain in f_1 score through semi-supervised updates, the red highlights zero-gain.

of labelled data to unlabelled data; the curves represent the average over 50 repeats. Semi-supervised learning consistently improves the classification performance, particularly for low proportions of labelled observations. Notably, at 2.49% labelled data, there is a 0.0380 improvement in the f_1-score, corresponding to a 3.87% reduction in the classification error.

For very low proportions of labelled data ($< 0.995\%$), semi-supervised learning can decrease the classification performance – shown by a negative gain in f_1-score (or error reduction) in Figures 6 and 7. It hypothesised that the performance drops for large quantities of unlabelled data ($m \gg n$), because the natural wighting in the log-likelihood leads to the labelled instances being effectively ignored [9, 15]. To accommodate for much larger sets of unlabelled data ($m \gg n$), a re-weighted version of the joint-likelihood has been suggested [9, 12]; the investigation of this approach is suggested for future work.

Intuitively, as the proportion of labelled data reaches 100% ($m \ll n$),

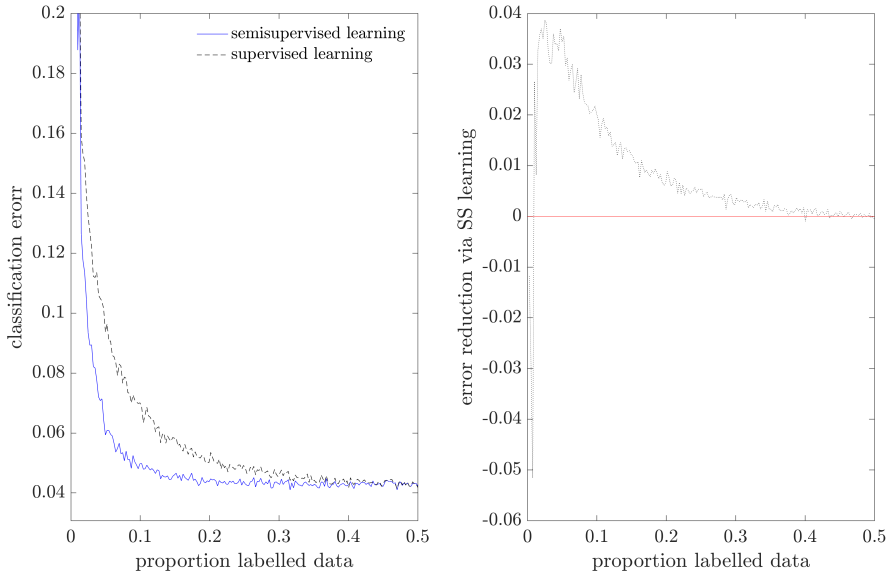


Figure 7: Classification error (e) for the supervised GMM vs. the semi-supervised GMM. Left: classification error for an increasing proportion of labelled data. Right: error reduction through semi-supervised updates, the red line highlights zero-error-reduction.

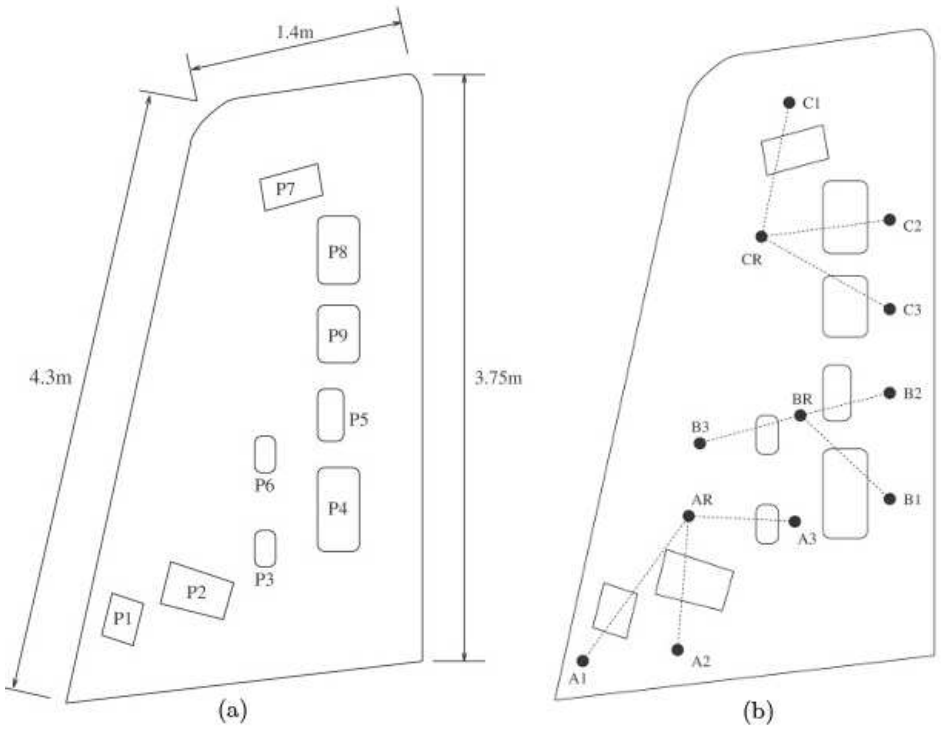
improvements through semi-supervised learning reduce, as there is less information gain from smaller sets of unlabelled signals. Considering the chosen method for density estimation, and the structure of the simulated data, these results are to be expected: as discussed, the underlying density is well approximated by the chosen mixture model (a GMM in this case, Figure 5b). The validity of this assumption is critical when using generative mixture models for semi-supervised learning.

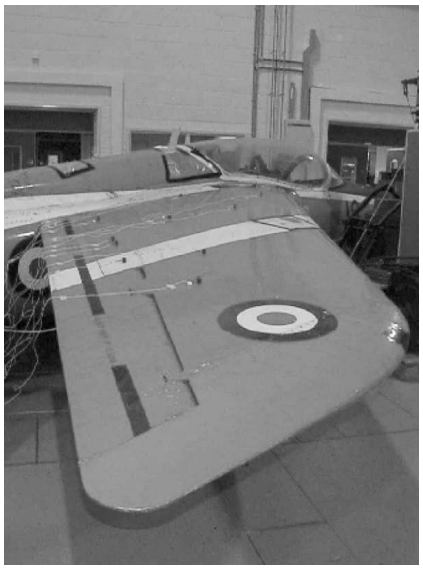
4.2. Gnat aircraft data

The Gnat data are an experimental dataset, concerning the wing of a Gnat aircraft [26]. During ground-vibration tests, the wing was excited using an electrodynamic shaker and band-limited white-noise. A network of sensors recorded the acceleration response at different points on the wing, shown in Figure 8b. During the experiments, artificial damage was introduced by sequentially removing one of nine inspection panels; the panels are shown in Figure 8a. (It is acknowledged that the removal of each panel represents a fairly large and significant fault.) The data represent a nine-class damage classification (location) problem; one class is associated with the removal of each panel.

The network of sensors are split into groups A, B and C; each group has one centrally-placed reference transducer (AR, BR, CR) and three response transducers (A/B/C1-3), labelled in Figure 8b. As with the simulated example, transmissibilities are used to monitor any changes that might relate to damage; specifically, the ratio of the response (transmitted) spectrum, to that of the reference spectrum. As such, there are nine transmissibilities – three for each group, represented by dotted lines in Figure 8b. In all cases 1024 spectral lines were recorded, from 1024 to 2048Hz [26].

The are 1782 observations for each transmissibility – 198 for each damage condition. To reduce the dimensionality of the dataset, each transmissibility is reduced to a single novelty index through a Mahalanobis-squared-distance (MSD) novelty detector [1, 26]. To build the novelty detectors, regions of spectral lines from each transmissibility are selected with the aid of a Genetic Algorithm (GA). Briefly, the GA iterates though a population of MSD novelty detectors, learnt with different sets of spectral lines. The *fitness* of each





(c)

Figure 8: Wing schematics: (a) panel locations, (b) sensor layout, (c) experimental setup.

set is assessed using the inverse classification error on a validation-set for a simple multilayer perception [10]. The 'fittest' sets are passed on to the next generation by combining their solutions. Mutation is also included by the occasional random switch of a feature. For a detailed discussion of the feature selection procedure, the reader is referred to [27].

In summary, the data represent a nine-class classification problem, concerning damage location. As such, the label space is $y_i \in \{1, \dots, 9\}$. The measured signals were converted to the frequency domain, to define nine transmissibilities; each transmissibility is then represented by a single novelty index, compressing the observation data to nine dimensions, thus $\mathbf{x}_i \in \mathbb{R}^9$.

Results

The same classification test-procedure (applied to the simulated data) is now applied to the Gnat data; results are shown in Figures 9 and 10. Again, semi-supervised updates through EM consistently improve the f_1-score and reduce the classification error, while, in this application, the data represent more practical SHM data. As with the simulated example, for very low proportions of labelled data $< 1.26\%$ ($m \gg n$), semi-supervised model updates decrease the predictive performance, as the effect of the unlabelled data appear to outweigh the labelled instances in the likelihood cost function. The general improvements through the semi-supervised GMM indicate that the experimental data can be (at least approximately) represented with a mixture of Gaussians; the maximum increase in the f_1-score is 0.0405, corresponding to a 3.83% reduction in the classification error for 2.94% labelled data.

For both tests, it is believed that semi-supervised improvements should increase if the data is approximated by some more flexible likelihood, i.e. $p(\mathbf{x}_i \,|\, \boldsymbol{\theta})$. A nonparametric representation, or a discriminative approach, would be a natural way to achieve this.

5. Conclusions and future work

An alternative method for semi-supervised learning, which utilises both labelled and unlabelled measurements, has been introduced to Structural Health Monitoring (SHM). The probabilistic approach impiments Expectation Maximisation (EM) over a generative mixture model, to improve

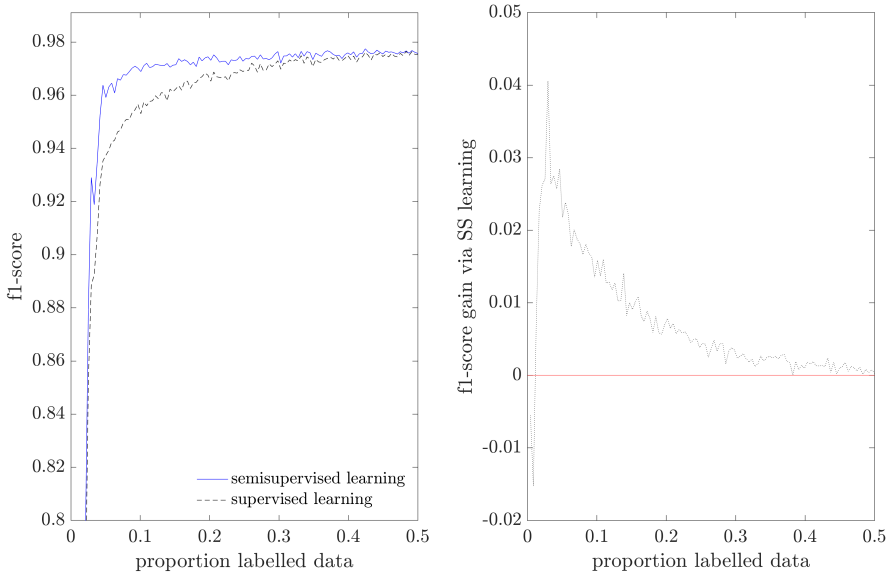


Figure 9: Classification performance assessed by the f_1-score for the supervised GMM vs. the semi-supervised GMM. Left: classification performance for an increasing proportion of labelled data. Right: the gain in f_1 score through semi-supervised updates, the red line highlights zero-gain.

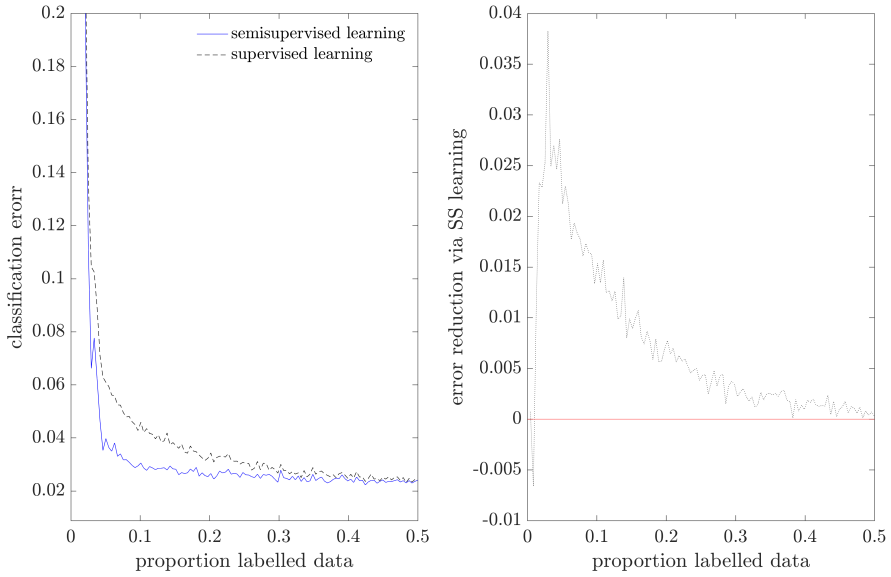


Figure 10: Classification error (e) for the supervised GMM vs. the semi-supervised GMM. Left: classification error for an increasing proportion of labelled data. Right: error reduction through semi-supervised updates, the red highlights zero-error-reduction.

the performance of damage classification under well-defined uncertainty – a significant advantage in risk-based applications. In the proposed method, a Gaussian Mixture Model (GMM) – learnt form *both* labelled and unlabelled measurements – is used to describe the underlying distribution of data from a simulated example and measured data from aircraft experiments (ground-tests). The classification accuracy (based on the GMM) is shown to improve significantly when the likelihood is maximised over the labelled *and* unlabelled data (semi-supervised learning), rather than the labelled data alone (supervised learning). More specifically, semi-supervised updates lead to 3.87% and 3.83% reductions in the classification error for the simulated and experimental datasets respectively. These improvements correspond to labelling just 2.49% of the measurements for the simulated data, and 2.94% of the measurements for the experimental data – low proportions of labelled data bring significant advantages to SHM, as investigating the structure to label the measured signals can be a high-cost procedure.

While the proposed method is successful, care must be taken to ensure that the assumed (parametric) mixture model – a GMM in this case – appropriately models the underlying distribution of data. If the imposed structure is inappropriate, the inclusion of unlabelled data will decrease the model quality. Considering this limitation, future work will apply the proposed semi-supervised methodology to nonparametric mixture models, in order to describe (more complex) underlying distributions of SHM data. Additionally, the proposed semi-supervised methodology should be incorporated within an online framework, such that streaming SHM signals – recorded from systems in operation – can be used to update the semi-supervised model of the data.

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.

References

[1] C. R. Farrar and K. Worden. *Structural Health Monitoring: a Machine Learning Perspective*. John Wiley & Sons, 2012.

[2] K. P. Murphy. *Machine Learning: a Probabilistic Perspective*. MIT press, 2012.

[3] A. Gelman, H. S. Stern, J. B. Carlin, D. B. Dunson, A. Vehtari, and D. B. Rubin. *Bayesian Data Analysis*. Chapman and Hall/CRC, 2013.

[4] D. Barber. *Bayesian Reasoning and Machine Learning*. Cambridge University Press, 2012.

[5] M. M. Moya, M. W. Koch, and L. D. Hostetler. One-class classifier networks for target recognition applications. In *Proceedings World Congress on Neural Networks*, pages 359–367, 1993.

[6] F. Schwenker and E. Trentin. Pattern classification and clustering: a review of partially supervised learning approaches. *Pattern Recognition Letters*, 37(1):4–14, 2014.

[7] L. Bull, K. Worden, G. Manson, and N. Dervilis. Active learning for semi-supervised structural health monitoring. *Journal of Sound and Vibration*, 437:373–388, 2018.

[8] L. Bull, T. Rogers, C. Wickramarachchi, E. Cross, K. Worden, and N. Dervilis. Probabilistic active learning: an online framework for structural health monitoring. *Preprint Submitted to Mechanical Systems and Signal Processing*, 2003.

[9] O. Chapelle, B. Scholkopf, and A. Zien. *Semi-Supervised Learning*. MIT press, 2006.

[10] B. C. M. *Pattern Recognition and Machine Learning*. Springer, 2006.

[11] T. Joachims. Transductive inference for text classification using support vector machines. In *Proceedings of the Sixteenth International Conference*

on Machine Learning, ICML '99, pages 200–209, San Francisco, CA, USA, 1999. Morgan Kaufmann Publishers Inc. ISBN 1-55860-612-2.

[12] K. Nigam, A. McCallum, S. Thrun, T. Mitchell, *et al.* Learning to classify text from labeled and unlabeled documents. *AAAI/IAAI*, 792:6, 1998.

[13] F. G. Cozman, I. Cohen, and M. C. Cirelo. Semi-supervised learning of mixture models. In *Proceedings of the 20th International Conference on Machine Learning (ICML-03)*, pages 99–106, 2003.

[14] A. P. Dempster, N. M. Laird, and D. B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal of the Royal Statistical Society: Series B (Methodological)*, 39(1):1–22, 1977.

[15] A. K. McCallumzy and K. Nigamy. Employing EM and pool-based active learning for text classification. In *Proc. International Conference on Machine Learning (ICML)*, pages 359–367. Citeseer, 1998.

[16] X. J. Zhu. Semi-supervised learning literature survey. Technical report, University of Wisconsin-Madison Department of Computer Sciences, 2005.

[17] S. Chen, F. Cerda, P. Rizzo, J. Bielak, J. H. Garrett, and J. Kovacevic. Semi-supervised multiresolution classification using adaptive graph filtering with application to indirect bridge structural health monitoring. *IEEE Transactions on Signal Processing*, 62(11):2879–2893, June 2014.

[18] S. Chen, F. Cerda, J. Guo, J. B. Harley, Q. Shi, P. Rizzo, J. Bielak, J. H. Garrett, and J. Kovacevic. Multiresolution classification with semi-supervised learning for indirect bridge structural health monitoring. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing*, pages 3412–3416, May 2013.

[19] A. Chebira, Y. Barbotin, C. Jackson, T. Merryman, G. Srinivasa, R. F. Murphy, and J. Kovavcevic. A multiresolution approach to automated classification of protein subcellular location images. *BMC Bioinformatics*, 8(1):210, Jun 2007.

[20] S. Mallat. *A Wavelet Tour of Signal Processing, Third Edition: The Sparse Way*. Academic Press, 3rd edition, 2008.

[21] D. J. MacKay. *Information Theory, Inference and Learning Algorithms*. Cambridge University Press, 2003.

[22] A. E. Bouzenad, M. El Mountassir, S. Yaacoubi, F. Dahmene, M. Koabaz, L. Buchheit, and W. Ke. A semi-supervised based k-means algorithm for optimal guided waves structural health monitoring: A case study. *Inventions*, 4(1), 2019.

[23] Y. Huang, L. Gong, S. Wang, and L. Li. A fuzzy based semi-supervised method for fault diagnosis and performance evaluation. In *2014 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, pages 1647–1651, July 2014.

[24] S. Theodoridis and K. Koutroumbas. *Pattern Recognition*. Academic Press, Boston, fourth edition edition, 2009.

[25] A. Papoulis. *Probabilities, Random Variables, and Stochastic Processes*. McGraw-Hill, 2965.

[26] G. Manson, K. Worden, and D. Allman. Experimental validation of a structural health monitoring methodology: Part III. damage location on an aircraft wing. *Journal of Sound and Vibration*, 259(2):365–385, 2003.

[27] K. Worden, G. Manson, G. Hilson, and S. Pierce. Genetic optimisation of a neural damage locator. *Journal of Sound and Vibration*, 309(3): 529–544, 2008.