Article:

On the application of domain adaptation in structural health monitoring

P. Gardner, X. Liu, K. Worden

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

A B S T R A C T

The application of machine learning within Structural Health Monitoring (SHM) has been widely successful in a variety of applications. However, most techniques are built upon the assumption that both training and test data were drawn from the same underlying distribution. This fact means that unless test data were obtained from the same system in the same operating conditions, the machine learning inferences from the training data will not provide accurate predictions when applied to the test data. Therefore, to train a robust predictor conventionally, new training data and labels must be recollected for every new structure considered, which is significantly expensive and often impossible in an SHM context. Transfer learning, in the form of domain adaptation, offers a novel solution to these problems by providing a method for mapping feature and label distributions for different structures, labelled source and unlabelled target structures, onto the same space. As a result, classifiers trained on a labelled structure in the source domain will generalise to a different unlabelled target structure. Furthermore, a holistic discussion of contexts in which domain adaptation is applicable are discussed, specifically for population-based SHM. Three domain adaptation techniques are demonstrated on four case studies providing new frameworks for approaching the problem of SHM.

© 2019 The Authors. Published by Elsevier Ltd. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

1. Introduction

Data-driven approaches to Structural Health Monitoring (SHM), specifically those utilising machine learning techniques, have achieved significant successes in a variety of applications [1–4]. The majority of these successes have involved supervised learning methods, meaning they require labelled training and test data. Moreover, whether supervised, unsupervised or semi-supervised, these techniques are developed on the assumption that both the training and test data are drawn from the same underlying distribution. This constraint typically means that machine learning approaches are specialised for a particular structure, application, and set of damage scenarios. Training and test data distributions for real-world operational structures may differ for several reasons: particular damage states may have only occurred in one of the two data sets, class distributions may vary over time due to operational conditions that may not be captured in the training set, training data may have been generated from a different structure from the test data etc. These issues prevent machine learning algorithms from generalising, leading to poor predictive performance. These problems hinder the algorithms in industry, as current approaches would require the collection of new labelled training data and the rebuilding of models for each individual structure, operational conditions and damage states, which is expensive and impractical for most scenarios. Supervised learning particularly suffers from these problems, as it requires labelled damage-state data for all considered damage scenarios,

---

which is often infeasible to obtain, nor is it economically viable for operational structures. These difficulties highlight a general desire to utilise information across a variety of structures and scenarios in order to produce machine learning approaches that generalise well.

From this discussion it is clear that a significant progressive step in SHM is developing an approach where information from across a population of structures can be used to perform inferences that generalise for the complete population (even if the number in the population is only two). This category of approach to SHM is defined as *population-based* SHM [5]. For example, in the case of a wind farm, there may be a variety of damage-labelled data across different wind turbines in the fleet, all subject to distinct operational conditions. It may be possible to have a specific wind turbine where labelled feature data is obtainable for particular normal and damage state conditions of interest, here called the *source* domain data. The operator may also have a different wind turbine of the same model with a large number of unlabelled data, called the *target* domain data. The data from these two turbines will have differences in their underlying distributions; these may come from the existence of inevitable manufacturing and assembling differences, or differing operational conditions due to their location within the farm. The question is, can the information from the labelled source domain data be used to create a method that generalises to the unlabelled target domain? This question motivates the work outlined within this paper.

It is also helpful at this point to define specific types of populations that exist within a population-based SHM context. The first main category are *homogeneous populations*. These form the simplest set of structures, in which each member of the population is nominally identical to the others, e.g. in terms of geometry, materials and topology. The differences in data distributions between members of this population type come from tolerances in manufacturing, local variations in material properties and small changes in operational conditions etc.; and therefore includes the example of the same model wind turbine within a fleet. The second category are *heterogeneous populations*, where differences in e.g., a combination of geometry, materials and topology, lead to significant differences in the data distributions, and potentially in the label classes. This group can be further divided into heterogeneous populations that are topologically similar or dissimilar, leading to consistency or inconsistency in labels related to location (stage two of Rytter's Hierarchy[1] [6]). An example of the first sub-division would be a population of three-span bridges, where one may have an overall span of 100m and another 50m. For the second subcategory, an example would be a five-span and a ten-span bridge, which poses a greater challenge for population-based SHM. It is clear that homogeneous populations are the simplest set of population type and therefore methods that apply to heterogeneous populations will apply to these scenarios.

Transfer learning, a subfield of machine learning, aims to improve a learner from one domain by transferring knowledge from a related domain [7]. Within transfer learning, there are a variety of approaches that differ in their aims and assumptions. *Domain adaptation* is a particular branch of transfer learning, where the focus is in reducing the distance between differing data distributions from source and target domains. This approach to transfer learning is therefore applicable to the SHM scenarios outlined above, namely when there is labelled data for a particular structure or operational environment and a set of unlabelled data for a different structure or operational environment. For these reasons, domain adaptation is investigated in this paper.

To demonstrate the novelty of this work, a review is provided of existing examples of transfer learning within the SHM literature. A majority of these studies have been about using transfer learning to improve image classification using deep convolutional neural networks. For example, Dorafshan et al. utilised the transfer learning mode within the AlexNet architecture for classifying cracks in images. A fine tuning approach was used to increase classification performance when compared to utilising the deep convolutional neural network as a conventional classifier [8]. Crack detection via image classification was also performed using a combination of deep convolutional neural networks and fine tuning transfer learning by Gao et al. [9]. In addition, Jang et al. implemented transfer learning to repurpose the GoogLeNet deep neural network architecture for automated crack detection in images for concrete structures [10]. These examples differ from the work provided here, as transfer learning within the convolutional neural network setting typically involves tuning the convolutional layer parameters from a network initially trained on a well-defined source domain.

Outside of deep learning, Chakraborty et al. derived a translated inductive transfer learning-based classifier, applying it to a fatigue damage case study [11]. The authors consider an example where sufficient labelled training data is only available from one of four sensors placed on an aluminium structure. Not only does this study use a different domain adaptation algorithm, it also does not consider the wider applications of domain adaptation within SHM, focusing on a sensor coverage problem. Finally, in the related field of non-destructive evaluation Ye et al. applied three domain adaptation methods in generating a robust hammering echo analysis technique for assessing concrete structures [12]. The algorithms assessed in this paper were transductive component analysis, geodesic flow kernel and maximum independence domain adaptation in combination with a Support Vector Machine (SVM), where the first two were found to provide better classification accuracies. These studies therefore do not provide a holistic discussion of contexts in which domain adaptation is applicable for SHM, neither do they outline the specific domain adaptation methods utilised in the paper, namely Transfer Component Analysis (TCA), Joint Domain Adaption (JDA) and Adaptation Regularization based Transfer Learning (ARTL).

The outline of this paper is as follows. Domain adaptation is mathematically introduced in Section 2 before specific algorithms are outlined, namely TCA, JDA and ARTL. The two proceeding sections discuss the applicability of these methods for two contexts with *heterogeneous populations*; nominally similar in topology, in Section 3, and where the topology is

---

[1] Rytter's Hierarchy states the problems within SHM in order of difficulty: detection, location, type, extent and prognosis.

considered different, in Section 4. Within each of these sections, two case studies are presented highlighting the potential of domain adaptation for addressing current issues in SHM. Finally, conclusions are presented.

2. Domain adaptation

Before defining transfer learning it is important to define two key objects: a *domain* and a *task* [7]:

- A *domain* $\mathcal{D} = \{\mathcal{X}, p(X)\}$ consists of a feature space $\mathcal{X}$ and a marginal probability distribution $p(X)$, where $X = \{\boldsymbol{x}_i\}_{i=1}^{N} \in \mathcal{X}$ i.e. a finite sample set from $\mathcal{X}$.
- A *task* for a given domain is defined as $\mathcal{T} = \{\mathcal{Y}, f(\cdot)\}$, where $\mathcal{Y}$ is a label space and $f(\cdot)$ is a predictive function (or can be consider as the conditional distribution $p(\boldsymbol{y}|X)$) learnt from a training data set $\{\boldsymbol{x}_i, y_i\}_{i=1}^{N}$, where $\boldsymbol{y} \in \mathcal{Y}$.

Using these definitions, transfer learning for a single source and target domain case can be defined as:

- *Transfer learning*: Given a source domain $\mathcal{D}_s$ and task $\mathcal{T}_s$ and target domain $\mathcal{D}_t$ and task $\mathcal{T}_t$, it is the process of improving the target predictive function $f_t(\cdot)$ in $\mathcal{T}_t$ using the knowledge learnt from $\mathcal{D}_s$ and $\mathcal{T}_s$, assuming $\mathcal{D}_s \neq \mathcal{D}_t$ and/or $\mathcal{T}_s \neq \mathcal{T}_t$.

Transfer learning methods are then based on whether $\mathcal{X}$, $p(X)$, $\mathcal{Y}$ or $p(\boldsymbol{y}|X)$ are consistent across the source and target [7]. Domain adaptation is one branch of transfer learning defined as:

- *Domain adaptation*: Given a source domain $\mathcal{D}_s$ and task $\mathcal{T}_s$ and target domain $\mathcal{D}_t$ and task $\mathcal{T}_t$, it is the process of improving the target predictive function $f_t(\cdot)$ in $\mathcal{T}_t$ using the knowledge learnt from $\mathcal{D}_s$ and $\mathcal{T}_s$, assuming $\mathcal{X}_s = \mathcal{X}_t$ and $\mathcal{Y}_s = \mathcal{Y}_t$, but that $p(X_s) \neq p(X_t)$ and $p(Y_s|X_s) \neq p(Y_t|X_t)$.

Domain adaptation is appropriate for scenarios in which the classifier will not generalise across the source and target domains due to differences in these distributions. As a result, methods for performing domain adaptation focus on minimising the distances (or divergence) between these distributions through some mapping $\phi(\cdot)$, such that $p(\phi(X_s)) \approx p(\phi(X_t))$ and $p(Y_s|\phi(X_s)) \approx p(Y_t|\phi(X_t))$. This requires the definition of a distance between two probability distributions; in transfer learning the Kullback-Leibler divergence [13], Jenson-Shannon divergence [14], Hellinger distance [15] and Maximum Mean Discrepancy (MMD) distance [16] have all been used in minimising differences between the source and target distributions.

In a population-based SHM context, domain adaptation is appropriate for scenarios in which the feature space and label space match i.e. $\mathcal{X}$ are both modal vectors relating to the same natural frequencies, and $\mathcal{Y}$ refer to the same location, type and extent of damage (although this is discussed further in Section 4). This fact makes domain adaptation most applicable for the homogeneous population context and for heterogeneous populations where both features and labels can be considered consistent i.e. when structures are topologically similar; although, as outlined in Section 4, it will be seen that these methods may be applied more broadly to more general heterogeneous populations as well.

In the following subsections, three domain adaptation methods are outlined, TCA, JDA and ARTL. These three techniques differ in their assumptions about consistencies between the source and target, with ARTL also seeking to align a hyperplane-based classifier in the transformed space.

Finally, before proceeding to define the methods, two classification metrics are outlined with which predictive performance of the domain adaptation techniques will be assessed. These metrics are constructed from the number of true positives (*TP*), false positives (*FP*), true negatives (*TN*) and false negatives (*FN*). The first metric is *accuracy* defined as,

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN}. \tag{1}$$

The second metric, the *macro F_1-metric* is formed from the precision P, and recall R, for each class $c \in \mathcal{Y}$,

$$P_c = \frac{TP_c}{TP_c + FP_c} \tag{2}$$

$$R_c = \frac{TP_c}{TP_c + FN_c} \tag{3}$$

and from these quantities, a class F_1-score and macro F_1-score are formed for each class $c \in \mathcal{Y}$,

$$F_{1,c} = \frac{2P_c R_c}{P_c + R_c} \tag{4}$$

$$F_{1macro} = \frac{1}{C} \sum_{c \in Y} F_{1,c} \tag{5}$$

where the macro-averaged F_1-metric equally weights the score for each class regardless of the proportion of data obtained within each class. This metric therefore provides a better reflection of classification performance when a small number of data points are in one class relative to another, as it weights all classes evenly.

2.1. Transfer component analysis

TCA assumes $p(X_s) \neq p(X_t)$ but that $p(Y_s|X_s) = p(Y_t|X_t)$ i.e. the marginal distributions are very different. The method seeks to learn a nonlinear transform from the feature space to a Reproducing Kernel Hilbert Space (RKHS) i.e. $\phi : \mathcal{X} \to \mathcal{H}$, where $p(\phi(X_s)) \approx p(\phi(X_t))$ and $p(Y_s|\phi(X_s)) = p(Y_t|\phi(X_t))$ [17]. The approach utilises the (squared) MMD distance as a criterion — the difference between two empirical means through a nonlinear mapping in a RKHS — where a kernel $k(\boldsymbol{x}_i, \boldsymbol{x}_j) = \phi(\boldsymbol{x}_i)^{\top}\phi(\boldsymbol{x}_j)$ is defined, leading to,

$$\text{Dist}\left(p\left(\tilde{X}_s\right), p\left(\tilde{X}_t\right)\right) = \left\|\frac{1}{N_s}\sum_{i=1}^{N_s}\phi(\boldsymbol{x}_{s,i}) - \frac{1}{N_t}\sum_{i=1}^{N_t}\phi(\boldsymbol{x}_{t,i})\right\|_{\mathcal{H}}^2 = \text{tr}(KM) \tag{6}$$

where $\tilde{X}$ are the transformed features, $K = \phi(X)^{\top}\phi(X) \in \mathbb{R}^{(N_s+N_t)\times(N_s+N_t)}$ given $X = X_s \cup X_t \in \mathbb{R}^{(N_s+N_t)\times D}$ where D is the dimension of the features, and M is the MMD matrix as defined by,

$$M(i,j) = \begin{cases} \frac{1}{N_s^2}, & x_i, x_j \in X_s \\ \frac{1}{N_t^2}, & x_i, x_j \in X_t \\ \frac{-1}{N_s N_t}, & \text{otherwise}. \end{cases} \tag{7}$$

Utilising the low-rank empirical kernel embedding $\tilde{K} = KWW^{\top}K$ [18], the distance can be rewritten as,

$$\text{Dist}\left(p\left(\tilde{X}_s\right), p\left(\tilde{X}_t\right)\right) = \text{tr}\left(W^T KMKW\right) \tag{8}$$

where $W \in \mathbb{R}^{(N_s+N_t)\times k}$ are the weights which perform a reduction and transformation.

This problem can now be formed in an optimisation framework, where by optimising the weights W, the marginal distributions are brought together in the transformed space. The problem is performed under a square Frobenius-norm regularisation constraint to control the complexity of W, and subject to kernel Principle Component Analysis (PCA) such that the trivial solution $W = 0$ is avoided as shown in the objective,

$$\min_{W^{\top}KHKW=\mathbb{I}} = \text{tr}(W^{\top}KMKW) + \mu\text{tr}(W^{\top}W) \tag{9}$$

where μ is a regularisation trade-off parameter, $H = \mathbb{I} - 1/(N_s + N_t)\mathbf{1}$ is a centring matrix, $\mathbb{I}$ is an identify matrix and $\mathbf{1}$ a matrix of ones. Using a Lagrangian approach the optimisation is converted to an eigenvalue problem which can be solved for W where the eigenvectors required for mapping correspond to the k smallest eigenvalues of,

$$(KMK + \mu\mathbb{I})W = KHKW\phi. \tag{10}$$

Finally the k-dimensional transformed feature space is calculated by $Z = KW \in \mathbb{R}^{(N_s+N_t)\times k}$. Once obtained, a classifier can be trained in the transformed space using the source data and subsequently implemented on the target data.

2.2. Joint domain adaptation

JDA assumes $p(Y_s, X_s) \neq p(Y_t, X_t)$ i.e. the joint distributions are very different. The method seeks to learn a nonlinear transform from the feature space to an RKHS i.e. $\phi : \mathcal{X} \to \mathcal{H}$, where $p(\phi(X_s)) \approx p(\phi(X_t))$ and $p(Y_s|\phi(X_s)) \approx p(Y_t|\phi(X_t))$ at the same time — given a joint distribution, which is the product of the marginal and conditional distributions [19].

However, the conditional in the target domain $p(Y_t|X_t)$ cannot be modelled directly as there are no labelled target data. To overcome this problem, JDA utilises a pseudo-labelling approach, whereby a classifier trained on the source data is applied to the target data in order to provide estimates of the labels $\hat{Y}_t$ — a naive form of semi-supervised learning. In addition, the posterior probabilities $p(Y|X)$ are difficult to obtain, meaning that JDA utilises the class-conditional distributions $p(X_s|Y_s)$ and $p(X_t|Y_t)$. By using the true source labels and pseudo target labels, JDA matches the conditional distribution for each class $p(X_s|Y_s = c)$ and $p(X_t|Y_t = c)$ where $c \in \{1, \ldots, C\}$ in the label set $\mathcal{Y}$.

The MMD between these class-conditional distributions (using the empirical kernel embedding) can be formed as,

$$\text{Dist}\left(p\left(\tilde{X}_s\right), p\left(\tilde{X}_t\right)\right) + \text{Dist}\left(p\left(Y_s|\tilde{X}_s\right), p\left(Y_t|\tilde{X}_t\right)\right) \approx \text{tr}\left(W^T KM_c KW\right) \tag{11}$$

noting that if $c = 0$ then this formulation becomes TCA and therefore if $c \in \{0, 1, \ldots, C\}$, both the marginal and class-conditionals distances (and hence an approximation of the joint) are minimised. As a result, the MMD matrix becomes,

$$M_c(i,j) = \begin{cases} \frac{1}{N_s^{(c)} N_s^{(c)}}, & x_i, x_j \in \mathcal{D}_s^{(c)} \\ \frac{1}{N_t^{(c)} N_t^{(c)}}, & x_i, x_j \in \mathcal{D}_t^{(c)} \\ \frac{-1}{N_s^{(c)} N_t^{(c)}}, & \begin{cases} x_i \in \mathcal{D}_s^{(c)} x_j \in \mathcal{D}_t^{(c)} \\ x_j \in \mathcal{D}_s^{(c)} x_i \in \mathcal{D}_t^{(c)} \end{cases} \\ 0, & \text{otherwise} \end{cases} \tag{12}$$

where $\mathcal{D}_s^{(c)} = \{x_i : x_i \in \mathcal{D}_s \wedge y(x_i) = c\}$ are the instances that belong in class c given the true source label $y(x_i)$ of x_i and $N_s^{(c)} = |\mathcal{D}_s^{(c)}|$; and $\mathcal{D}_t^{(c)} = \{x_i : x_i \in \mathcal{D}_t \wedge \hat{y}(x_i) = c\}$ are the instances that belong in class c given the pseudo-target label $\hat{y}(x_i)$ of x_i and $N_t^{(c)} = |\mathcal{D}_t^{(c)}|$ (where $\wedge$ is the logical AND symbol). Following the same formulation as TCA the optimisation problem (subject to the regularisation constraint and kernel PCA) again becomes an eigenvalue problem where the optimal W is obtained from the eigenvectors corresponding to the k smallest eigenvalues from,

$$\left(K \sum_{c=0}^{C} M_c K + \mu \mathbb{I} \right) W = KHKW\phi. \tag{13}$$

Due to the pseudo-labelling of the target features, [19] recommends running several iterations of the optimisation to find the optimal W. Again the k-dimensional transformed feature space is calculated by $Z = KW \in \mathbb{R}^{(N_s + N_t) \times k}$, and a classifier trained on the transformed source data can be applied to the transformed target data.

2.3. Adaptation regularisation-based transfer learning

ARTL incorporates a hyperplane-based classifier within the domain adaptation procedure, aiming to boost classification performance. The method makes the same assumptions as JDA, i.e. $p(Y_s, X_s) \neq p(Y_t, X_t)$ i.e. the joint distributions are very different; however, it also considers the discriminative directions between the domains via manifold regularisation [20].

The general framework of ARTL is constructed from a structural risk minimisation principle and regularisation theory. The optimisation objective function,

$$f = \min_{f \in \mathcal{H}} \sum_{i=1}^{N} \mathcal{L}(f(x_i), y_i) + \sigma ||f||_K^2 + \mu \text{Dist}(p(Y_s, \phi(X_s)), p(Y_s, \phi(X_s))) + \gamma MR(p(\phi(X_s))), p(\phi(X_t)) \tag{14}$$

is comprised of three components: minimisation of structural risk on the labelled source domain data, minimisation of the differences between the joint distributions in the source and target domains, and maximising the manifold consistency underlying the marginal distributions in the source and target domain.

This equation is optimised to find the predictive function $f = w^T \phi(x)$, where w are classifier parameters. The first two terms in Eq. (14) relate to the structural risk, where $\mathcal{L}(f(x_i), y_i)$ is a loss function that quantifies the fitness of f in predicting the training labels. This loss function is subject to a squared-norm regularisation constraint in $\mathcal{H}, \sigma ||f||_K^2$ where σ is the shrinkage regularisation parameter.

The second part of Eq. (14) performs joint domain adaptation, i.e. minimising $\text{Dist}(p(Y_s, \phi(X_s)), p(Y_s, \phi(X_s)))$. This structure forms similar equations to Eq. (11), however the distances are now in reference to the classifier such that,

$$\text{Dist}(p(Y_s, \phi(X_s)), p(Y_s, \phi(X_s))) \approx \text{tr}(\alpha^T K M_c K \alpha) \tag{15}$$

where M_c is as defined in Eq. (12) using $c \in \{0, 1, \ldots, C\}$. However, unlike TCA and JDA, α are classifier parameters such that $w = \sum \alpha_i \phi(x_i)$ and $f(x) = \sum \alpha_i K(x_i, x)$ (using the representer theorem [21]). As with TCA and JDA, a regularisation constraint is incorporated, where μ is the regularisation parameter. Pseudo-labelling is also utilised, where the predictive function is used to classify the unlabelled target data, where these are used to update M_c iteratively.

The third aspect of ARTL is manifold regularisation governed by γ. This tries to make use of the unlabelled target domain data via the marginal distributions in the source and target domain. The assumption here is that the conditional distribution of two points in the source and target domains will be similar if those points are close in terms of the intrinsic geometry of the marginal distributions, known as the *manifold assumption* [22]. This leads to the following manifold regularisation term,

$$MR(p(\phi(X_s)), p(\phi(X_t))) = \sum_{i,j=1}^{N_s + N_t} \left(f(x_i) - f(x_j) \right)^2 W_{i,j} = \sum_{i,j=1}^{N_s + N_t} f(x_i) L_{i,j} f(x_j) \tag{16}$$

where W is the *graph affinity matrix* defined as,

$$W(i,j) = \begin{cases} \cos(\boldsymbol{x}_i, \boldsymbol{x}_j), & \boldsymbol{x}_i \in \mathcal{N}_p(\boldsymbol{x}_j) \vee \boldsymbol{x}_j \in \mathcal{N}_p(\boldsymbol{x}_i) \\ 0, & \text{otherwise} \end{cases} \tag{17}$$

where $\mathcal{N}_p(\boldsymbol{x}_i)$ is the set of p-nearest neighbours of points $\boldsymbol{x}_i$ and L is the *normalised graph Laplacian matrix*,

$$L = \mathbb{I} - D^{-1/2}WD^{-1/2} \tag{18}$$

where $D_{i,i} = \sum_{j=1}^N W_{i,j}$. Manifold regularisation is then enforced in terms of $\boldsymbol{\alpha}$,

$$MR(p(\phi(X_s)), p(\phi(X_t))) = \text{tr}(\boldsymbol{\alpha}^T KLK\boldsymbol{\alpha}) \tag{19}$$

By substituting a machine learning loss function into Eq. (14), an adaptive prediction function f can be inferred. The authors in [20] demonstrate ARTL for a hinge-loss and least-squares cost function. Following [20], the objective function for a regularised least-squares approach becomes,

$$\boldsymbol{\alpha} = \min_{\boldsymbol{\alpha} \in \mathbb{R}^N} ||(\boldsymbol{y} - \boldsymbol{\alpha}^T K)E||_F^2 + \text{tr}(\sigma\boldsymbol{\alpha}^T K\boldsymbol{\alpha} + \boldsymbol{\alpha}^T K(\mu M_c + \gamma L)K\boldsymbol{\alpha}) \tag{20}$$

where E is a diagonal label indicator matrix, i.e. $E_{i,i} = 1$ if $\boldsymbol{x}_i \in \mathcal{D}_s$ and zero otherwise, such that only the labelled source domain is considered in the loss function. The solution to Eq. (20) is subsequently formed by setting the derivative of the objective function to zero,

$$\boldsymbol{\alpha} = ((E + \mu M_c + \gamma L)K + \sigma\mathbb{I})^{-1}EY^T. \tag{21}$$

As previously stated, the adaptive classifier can then be formed from $f(\boldsymbol{x}) = \sum \alpha_i K(\boldsymbol{x}_i, \boldsymbol{x})$.

3. Population-based SHM case studies

Two case studies are presented in this section demonstrating the applicability of domain adaptation, specifically TCA, JDA and ARTL, for the subclass of *heterogeneous populations* where two structures are defined as similar topologically (in this case given their lumped-mass structure). These studies address two specific problems in SHM. The first problem deals with multi-site damage location, showing the method's applicability to multi-class classification contexts. Here, two numerical three-storey structures are utilised in generating data, where the structures differ in geometric dimensions and material properties. The second case study considers a scenario in which a numerical simulation is used as the source domain for a two-class damage detection problem, where the target domain is data from an experimental structure. This presents the effectiveness of the approaches in generating damage state labels when it would otherwise have been infeasible or not economically viable to obtain.

3.1. Representative three storey building structure

In both the scenarios considered here, the populations are a collection of three-storey shear structures, where the numerical structures are as depicted in Fig. 1, i.e. a lumped-mass model of a shear structure in bending. For each numerical simulation, the three masses $\{m_i\}_{i=1}^3$ are calculated from a given rectangular volume $v = l_m w_m t_m$ where the density ρ at each mass is considered different and l_m, w_m, t_m are the length, width and thickness of the rectangular volume. Similarly the three stiffness coefficients $\{k_i\}_{i=1}^3$ are composed from the tip stiffness of four cantilever beams in parallel i.e. $4k_b = 4(3EI/l_b^3)$, where E is the elastic modulus, I the second moment of area and l the beam length. Each stiffness coefficient is constructed from a different E. Additionally, damping coefficients $\{c_i\}_{i=1}^3$ are given for each structure but are not derived from a physical model.

For each case study, the damage scenario under consideration is an open crack. Specifically, a crack of length l_{cr} is introduced into one of the four beams at a particular degree-of-freedom. For example if the crack is located between the fixed support and first mass then $k_1 = 3k_b + k_d$; where k_d is the tip stiffness of a cantilever beam with a crack length l_{cr} at location l_{loc} along the length of the beam. In this paper the stiffness reduction due to an open crack in a cantilever beam is modelled as proposed by Christides and Barr in [23]. This model adopts a function of elastic modulus and second moment of area across the length of the beam x given by,

$$EI(x) = \frac{EI_0}{1 + C\exp(-2\alpha|x - l_{loc}|/t_b)} \tag{22}$$

where I_0 is the second moment of the undamaged beam, t_b the thickness of the beam and α a coefficient experimentally defined by Christides and Barr as 0.667. The constant $C = (I_0 - I_c)/I_c$ is a function of the undamaged I_0 and damaged second moments of area I_c, which for a rectangular beam are $I_0 = (w_b t_b^3)/12$ and $I_c = (w_b - l_{cr})t_b^3/12$. The damaged tip stiffness k_d is obtained from $k_d = -F/y_{tip}$ where F is a given force and y_{tip} is the tip deflection from numerically integrating the Euler-Bernoulli bending beam equation,
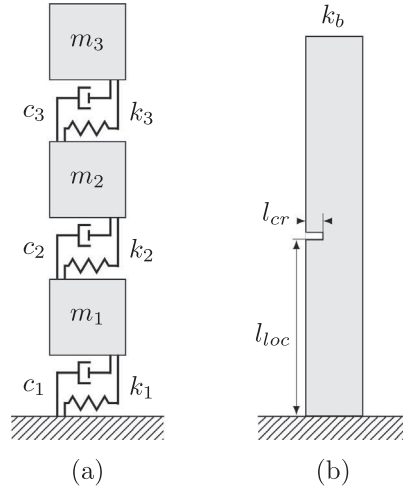
Fig. 1. Schematic of the three degree-of-freedom shear structure: panel (a) is of the full system and panel (b) depicts the cantilever beam component where $\{k_i\}_{i=1}^3 = 4k_b$ i.e. the stiffness coefficients in (a) are generated from four times the tip bending stiffness in (b).

$$\frac{\partial^2 y}{\partial x^2} = -\frac{M(x)}{EI(x)} \tag{23}$$

The damped natural frequencies $\{\omega_{d,i}\}_{i=1}^3$ and damping ratios $\{\zeta_i\}_{i=1}^3$ are subsequently calculated given these mass, damping and stiffness values, and used as features in the following classification tasks.

3.2. Multi-class case study

The first case study considers a multi-class classification problem between two different three-storey building structures. The SHM scenario is a three-class detection and location problem, i.e. for each system, an undamaged class, labelled '1', and two damaged classes at different locations, labelled '2' and '3', exist. It is assumed that these labels are known for the source structure and unknown for the target structure. Domain adaptation is implemented in order to build a classifier using the labelled information in the source domain that holds for the target domain. In order to demonstrate the effectiveness of TCA and JDA, a k-Nearest Neighbour (k-NN) classifier is used; as if the learnt mapping is accurate, the source and target distributions should coincide, and therefore be close in Euclidean distance. ARTL on the other hand, is a reconstruction of a hyperplane-based classifier, where here the regularised least-squares form is implemented.

The two damage classes refer to an open crack located at the first and third degrees-of-freedom, i.e. k_1 and k_3 respectively. The crack is considered to be at the midpoint along the length of one of the four beams at the particular degree-of-freedom. The crack lengths are defined as 50% of the beam width for each structure, such that the damage labels are comparable between the two structures.

The source and target structures are steel and aluminium three-storey building structures, where the geometric dimensions are different. The population is thus heterogeneous, the structures are not 'nominally identical'; however, they are topologically similar, in that a simple change of material or geometry parameters transforms one into the other. Interestingly, this shows that 'topologically equivalent' means close to the bounds between homogeneous populations and heterogeneous populations. The material and geometric properties for the source and target structures are displayed in Table 1. The elastic modulus, density and damping coefficients are set as probability distributions such that $\{m_i, c_i, k_i\}_{i=1}^3$ are obtained from a random draw from these distributions.
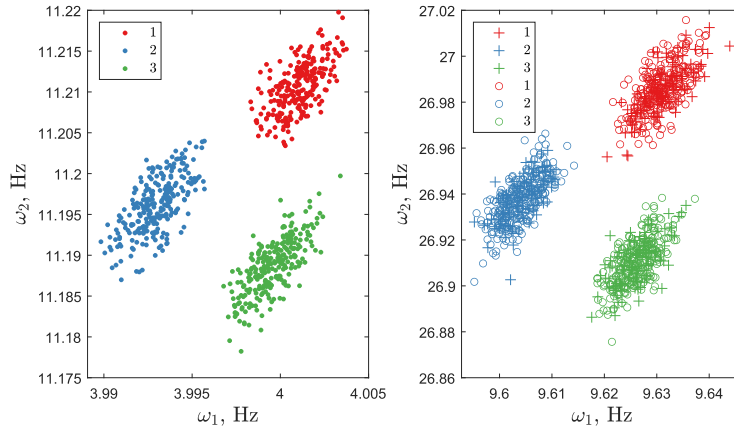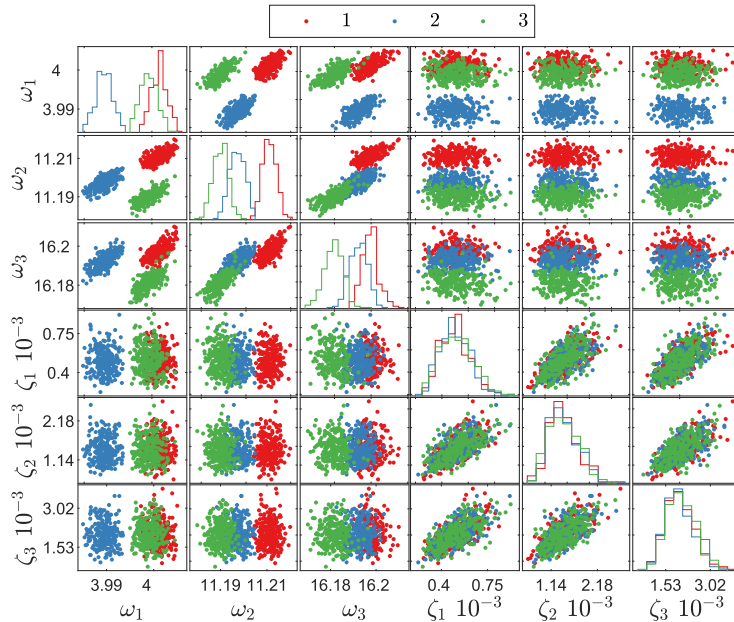
Following the approach outlined in Section 3.1, the damped natural frequencies and damping ratios are obtained for each of the three classes. Consequently, these quantities are utilised as features i.e. $X_s \in \mathbb{R}^{N_s \times 6}$ and $X_t \in \mathbb{R}^{N_t \times 6}$. In order to visualise these high-dimensional feature spaces two-dimensional comparisons are presented. Fig. 2 demonstrates a comparison of the first (ω_1) and second (ω_2) natural frequencies for the source and target domain showing the differences in magnitude between the source and target domain. This depiction forms part of Figs. 3 and 4 (i.e. second row from the top and first column), where each quadrant in Figs. 3 and 4 is a two-dimensional comparison between feature combinations, ordered natural frequencies followed by damping ratios i.e. the bottom-left quadrant is the first natural frequency against the third damping ratio. (It is noted that this format for displaying features spaces in figures is used throughout this paper).

For this case study, the training data used to infer the mappings in the domain adaptation methods were 250 repeats for each class from the labelled source domain and 100 repeats for each class from the unlabelled target domain i.e. $N_s = 750$ and $N_t = 300$. The test data were comprised of an additional 250 repeats for each class from the target domain (denoted by

Table 1
Properties of the source and target structures for the multi-class case study.

Property	Unit	Source	Target
Beam geometry, $\{l_b, w_b, t_b\}$	mm	$\{300, 40, 8\}$	$\{160, 25, 6\}$
Mass geometry, $\{l_m, w_m, t_m\}$	mm	$\{400, 400, 40\}$	$\{300, 250, 25\}$
Crack geometry, $\{l_{cr}, l_{loc}\}$	mm	$\{20.0, 150\}$	$\{12.5, 80\}$
Elastic modulus, E	GPa	$\mathcal{N}\left(210, 1 \times 10^{-9}\right)$	$\mathcal{N}\left(71, 1 \times 10^{-10}\right)$
Density, ρ	kg/m³	$\mathcal{N}(7800, 50)$	$\mathcal{N}(2700, 10)$
Damping coefficient, c	Ns/m	$\mathcal{G}(8, 0.8)$	$\mathcal{G}(50, 0.1)$

Fig. 2. Comparison of the first two dimensions, ω_1 and ω_2, for the source (left) and target (right) domain features for the multi-class case study. The training and testing data for the target domain are denoted by $(+)$ and $(\circ)$ respectively.

Fig. 3. Source domain features for the multi-class case study (where natural frequencies are in Hz).

$(\circ)$ in Fig. 4), i.e. $N_{test} = 750$. These figures demonstrate the differences between both domains, not only in absolute values but also in the size and scaling. To demonstrate and motivate the need for domain adaptation in this example, a k-NN classifier (using Euclidean distance and $k = 2$) was trained on the source domain data and applied to the unlabelled training and testing target domain data (where the features were normalised to the source domain for computational reasons). As
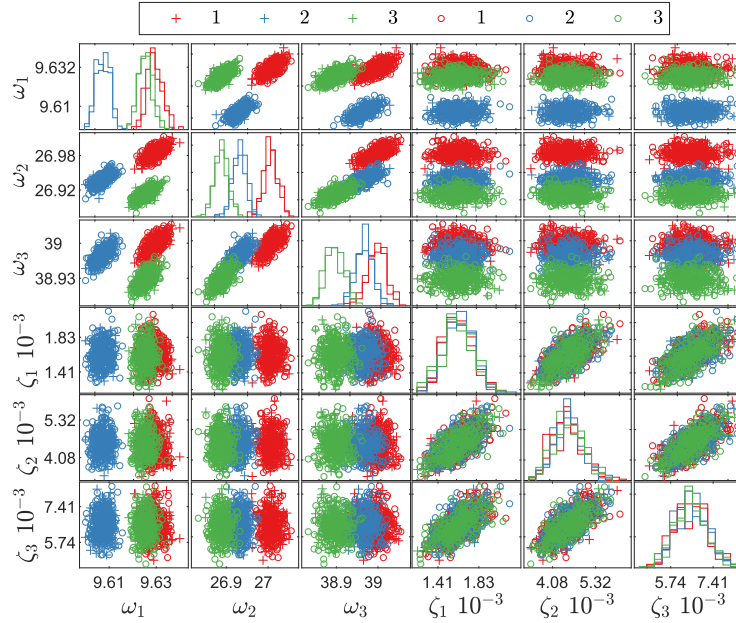
Fig. 4. Target domain features for the multi-class case study (where natural frequencies are in Hz). The training and testing data are denoted by (+) and (○) respectively.

expected, the k-NN classifier without domain adaptation, fails to correctly label the data for both scenarios as demonstrated by macro F_1-scores and accuracies of 0.167 and 33.3% for both the training and testing target domain.

3.2.1. Domain adaptation

The three domain adaptation methods were applied to the multi-class classification data in order to infer a general classifier for both the source and target domains. Cross-validation was performed, using the macro F_1-score for TCA and JDA, and the regularised square loss for ARTL as the cost function. Using a k-fold approach, with ten folds, the number of components and regularisation parameters were identified (this involves partitioning the data into a number of training and validation data sets, where the parameters with the best average metric over the data sets are selected). For each method, a linear kernel was utilised.

Cross-validation identified four components and $\mu = 0.5$ for TCA. Fig. 5 presents a visualisation of the learnt mapping where it can be seen that the first component has not consistently mapped both the source and target domains. This result is reflected in Fig. 6, where the marginal distributions for each component are presented. Given the aim of TCA, it is expected that the source and target domain marginal distributions should be close; however, this is not the case for the first component Z_1.

JDA was implemented with ten iterations and a k-NN classifier for determining pseudo labels (in order to maintain a consistent comparison). Cross-validation identified five components and $\mu = 1$ for JDA. A visualisation of the mappings is presented in Fig. 7, showing a similar problem in the first component as in TCA. Fig. 8 displays the marginal and class-conditional distributions for each component. Visually, it can be seen that the second, fourth and fifth components have marginal and class-conditional distributions that match for the source and target domains. For the first and third components, the target domain is approaching the source domain class-conditional distributions, explaining the behaviour in the marginal distributions. This has the unintended benefit of providing greater separability in the target domain.

ARTL is not visualised here as the α parameters are no longer weights solely associated with the transformed domain, as in TCA and JDA, but are also affected by the classifier and manifold. Cross-validation, via minimising the regularised square loss, selected one transfer component, $\sigma = 0.1$, $\mu = 0.001$, $\gamma = 0.1$ and one nearest neighbour; ten iterations were also implemented.

Classification results on the target domain for the multi-class case study are presented for the three approaches in Table 2; where the source domain is used as training data for the classifier and both target domain data sets provide two testing scenarios. Several conclusions can be drawn from these results. Firstly, domain adaptation techniques are applicable to multi-class scenarios, and therefore can be applied to the complete levels of Rytter's hierarchy. Secondly, all three domain adaptation methods significantly outperform a k-NN classifier trained on the untransformed data. Additionally, JDA outperformed TCA, highlighting that the assumption that $p(Y_s|\phi(X_s)) = p(Y_t|\phi(X_t))$ does not hold, as expected for a heterogeneous population. ARTL shows increased classification performance, demonstrating the benefit in aligning the classifier hyperplane via manifold regularisation.
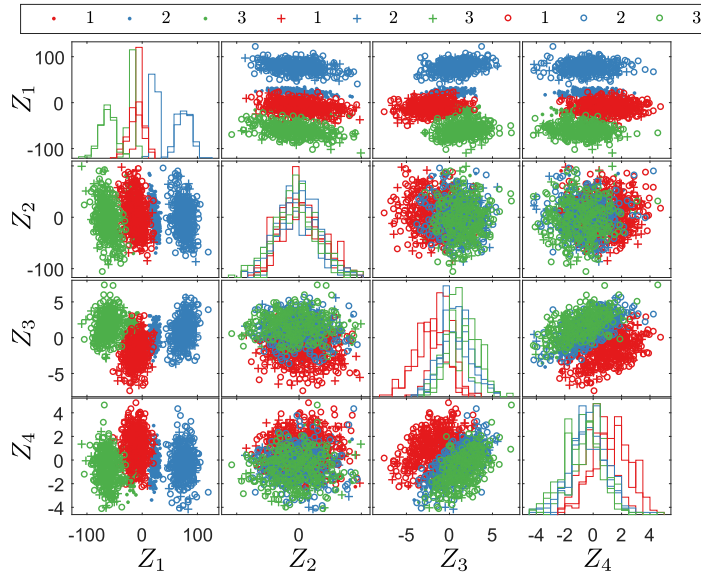
Fig. 5. A visualisation of the TCA-transformed components for the multi-class case study. The source domain and the training and testing data for the target domain are denoted by $(\cdot), (+)$ and $(\circ)$ respectively.
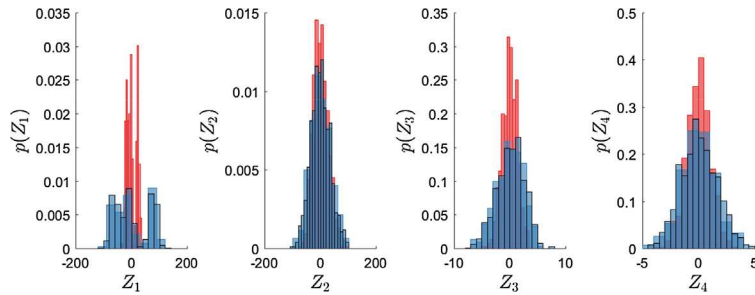

Fig. 6. The marginal distributions for each TCA-transformed component for the multi-class case study. The source domain and target domain results are red and blue respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

3.3. Numerical-to-experimental domain adaptation

The second topologically-similar case study involves utilising a numerical simulation in generating labelled source domain data for a two-class damage detection problem. In this scenario the unlabelled target domain data are from an experimental structure, shown in Fig. 9. The case study highlights that domain adaptation offers a method for generating damage state labels from numerical models that will apply to an operational structure. This capability has the potential to make supervised learning approaches feasible across industrial applications where damage state data are unobtainable. These techniques therefore offer a novel framework for utilising physics-based models in SHM, outside of traditional model updating [24].

The source domain data from the numerical structure is generated as outlined in Section 3.1. The simulated damage scenario was a 17.5mm, midpoint crack in one of the four beams at the first degree-of-freedom, k_1. The properties of the numerical structure are outlined in Table 3; where the structure and crack geometries are equivalent to the experimental structure, and the material properties are typical for aluminium 6082. Clearly, the numerical model is an oversimplification of the experimental structure given that it only includes the bending stiffness of the beams, excluding shear stiffness, full geometric features and modelling of the bolted joints etc. However, this highlights the applicability of using labels generated from physics-based models in domain adaptation. Typically physics-based models will involve simplifications, as all physics cannot generally be modelled, meaning that model-form errors will exist. In addition, it will often be impossible to validate the physics-based model, as damage state data from the operational structure is unobtainable, as previously stated. 250 repeats for each class were obtained from randomly sampling the properties in Table 3 i.e. $N_s = 500$.

Target domain data were collected via modal testing using an electrodynamic shaker attached to the first floor of the structure, as depicted in Fig. 9. The experimental structure, constructed from aluminium 6082, was tested in both the
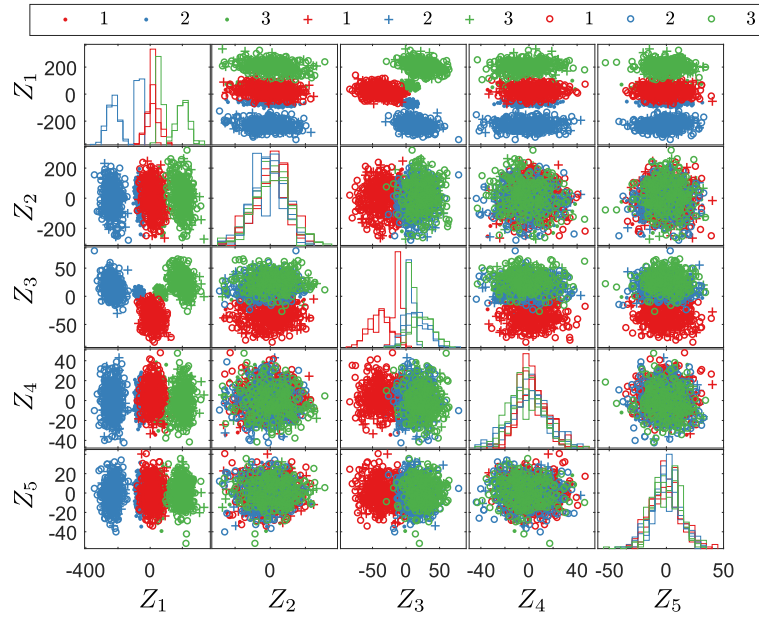
Fig. 7. A visualisation of the JDA transformed components for the multi-class case study. The source domain and the training and testing data for the target domain are denoted by $(\cdot), (+)$ and $(\circ)$ respectively.
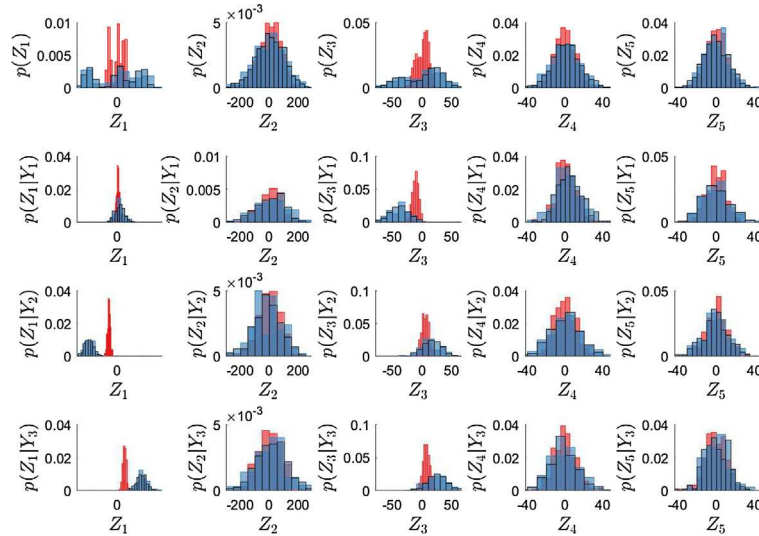

Fig. 8. The marginal and class-conditional distributions for each JDA-transformed component for the multi-class case study. The source domain and target domain results are red and blue respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 2
Classification results for the multi-class case study trained on the labelled source domain and applied to the unlabelled target domain.

Method		k-NN	TCA	JDA	ARTL
Mapping Training	Accuracy	33.3%	85.3%	93.3%	99.7%
	F_1-score	0.167	0.843	0.931	0.997
Mapping Testing	Accuracy	33.3%	84.8%	93.7%	99.6%
	F_1-score	0.167	0.838	0.936	0.996

Fig. 9. Experimental setup of the three-storey building structure.

undamaged and damaged condition, where damage was introduced as a saw cut of 17.5mm on the midpoint of the front-right beam of Fig. 9. The structure was excited with a 6553.6Hz broadband white-noise excitation containing 16384 spectral lines (0.2Hz resolution) with a Hanning window on the force excitation and acceleration response. Five repeats were obtained for each damage class i.e. $N_t = 10$.

The features utilised in this scenario were the first three bending natural frequencies meaning $X_s \in \mathbb{R}^{N_s \times 3}$ and $X_t \in \mathbb{R}^{N_t \times 3}$, where Fig. 10 displays the feature sets for the source and target domain. It can be clearly seen that the natural frequencies are underestimated by the numerical model, by approximately a factor of two, due to the oversimplifications in the model. This case presents a more challenging problem for the domain adaptation techniques, highlighting their applicability when physics-based models have not been validated and may contain model-form errors, but capture to some extent the changes in the features due to damage.

Classification, for all approaches but ARTL, was performed using a k-NN classifier for the same reason as the multi-class case study, with $k = 3$ and using Euclidean distance. ARTL was performed using the regularised least-squares cost function. A k-NN classifier, trained on the (normalised) source domain and applied to these (normalised) target domain features produces a macro F_1-score of 0.333 and an accuracy of 50%. This again highlights the need for performing domain adaptation.

3.3.1. Domain adaptation

All three domain adaptation techniques were applied in the numerical-to-experimental domain adaptation scenario. Five-fold cross-validation was performed via maximising the macro F_1-score for TCA and JDA, and minimising the regularised squared loss for ARTL. A linear kernel was implemented for each approach.

A visualisation of the classification results for TCA, and the inferred transfer components, is presented in Fig. 11. Five-fold cross-validation selected two transfer components and $\mu = 5 \times 10^{-4}$. The target domain data have been correctly classified for all but a single data point, using a k-NN classifier trained on the source domain data and applied to the target domain data. The marginal distributions in Fig. 12 visualises that the source and target domain components are close together.

Table 3
Properties of the source structure for the numerical-to-experimental domain adaptation case study.

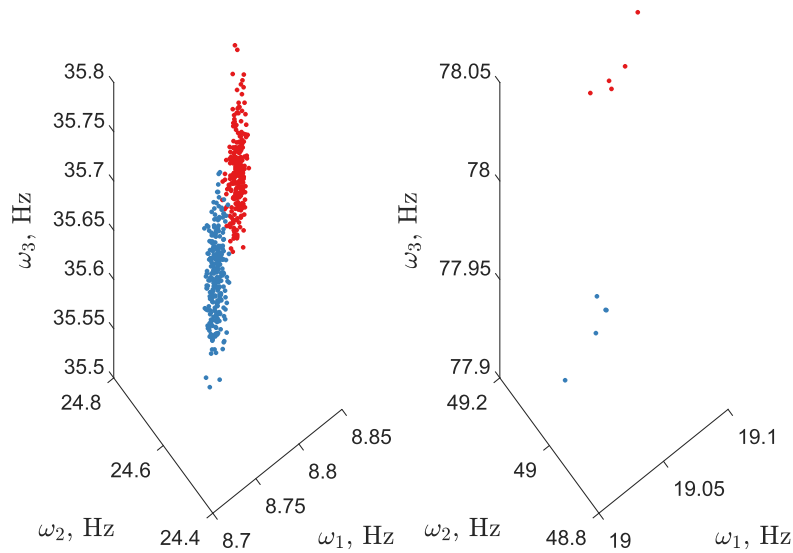| Property	Unit	Source
Beam geometry, $\{l_b, w_b, t_b\}$	mm	$\{177.8, 25.4, 6.35\}$
Mass geometry, $\{l_m, w_m, t_m\}$	mm	$\{304.8, 254.0, 25.4\}$
Crack geometry, $\{l_{cr}, l_{loc}\}$	mm	$\{17.5, 88.9\}$
Elastic modulus, E	GPa	$\mathcal{N}\left(71, 1 \times 10^{-9}\right)$
Density, ρ	kg/m^3	$\mathcal{N}(2700, 50)$
Damping coefficient, c	Ns/m	$\mathcal{G}(9, 0.5)$

Fig. 10. Source (left) and target (right) domain features for the numerical-to-experimental domain adaptation case study; showing both the undamaged (red) and damaged (blue) labels. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
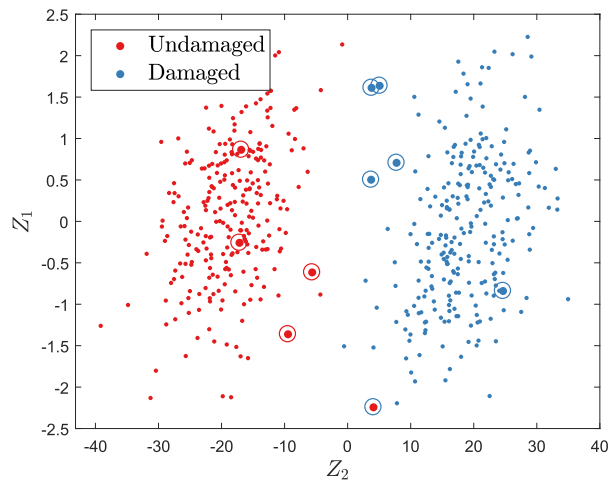

Fig. 11. Classification results for TCA applied to the numerical-to-experimental domain adaptation case study; showing labelled source domain data ($\cdot$), the true labelled target domain data ($\bullet$), and the predicted target domain labels ($\bigcirc$).
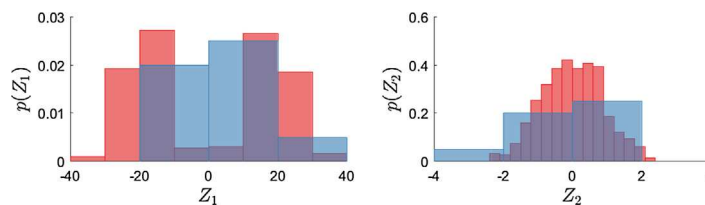

Fig. 12. The marginal distributions for each TCA-transformed component for the numerical-to-experimental domain adaptation case study. The source domain and target domain results are red and blue respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
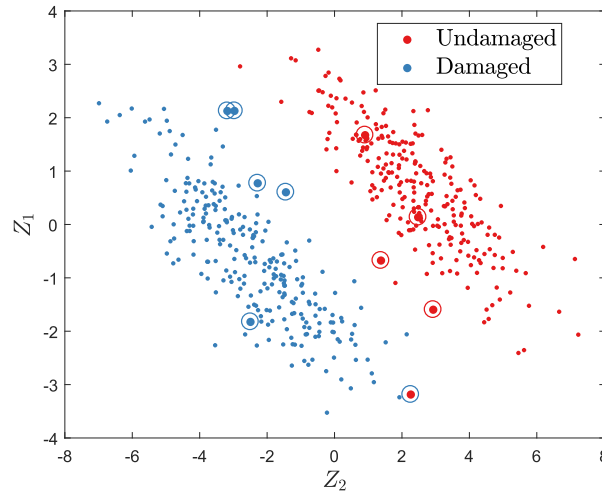
Fig. 13. Classification results for JDA applied to the numerical-to-experimental domain adaptation case study; showing labelled source domain data (·), the true labelled target domain data (•), and the predicted target domain labels (◯).

The visualisation of the JDA classification results, shown in Fig. 13, again shows classification of all but one data point successfully. In addition, JDA performs a better rotation of the source and target domain data such that the marginal and class-conditional distributions are closer together, as demonstrated in Fig. 14. These results likely show that there is added benefit in accounting for differences in the joint distributions between the source and target domains, although in terms of classification accuracy, JDA produces the same results as TCA meaning that there is not enough data to judge conclusively. Again five-fold cross-validation was applied, where two components and $\mu = 1$ were selected.

ARTL parameters were identified via cross-validation, where the cost function was the regularised squared loss. The parameters selected were one transfer component, $\sigma = 0.01$, $\mu = 0.01$, $\gamma = 1 \times 10^{-3}$ and one nearest neighbour; ten iterations were also implemented.

A summary of the classification results for the numerical-to-experimental domain adaptation case study are shown in Table 4. TCA and JDA both correctly classify all but one data point in the target domain, showing their applicability when utilising physics-based models in generating labels for operational structures. These results show that these methods can provide a way of alleviating problems associated with a lack of available damage state data. In contrast, ARTL completely fails to infer the decision bound; likely due to difficulties in inferring a hyperplane-based classifier from very few data points in the target domain. This problem will be compounded when cross-validation is used to infer the parameters of ARTL, as poor performance in one fold may dramatically effect the selection of parameters.
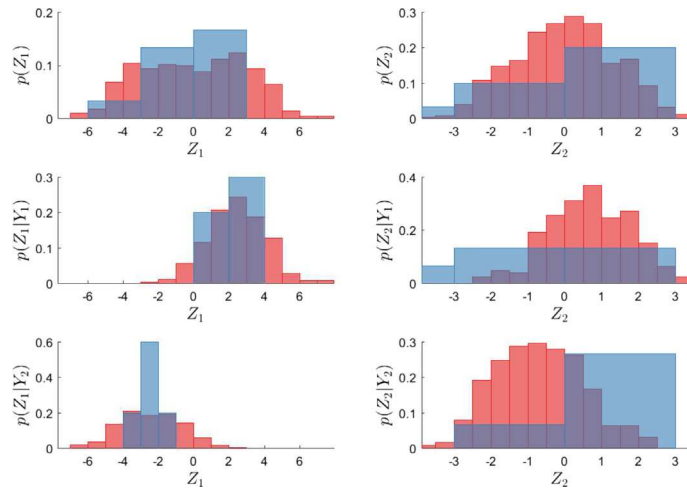

Fig. 14. The marginal and class-conditional distributions for each JDA-transformed component for the numerical-to-experimental domain adaptation case study. The source domain and target domain results are red and blue respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 4
Classification results for the numerical-to-experimental domain adaptation case study trained on the labelled source domain and applied to the unlabelled target domain.

Method	k-NN	TCA	JDA	ARTL
Accuracy	50.0%	90.0%	90.0%	50.0%
F_1-score	0.333	0.899	0.899	0.333

4. Truly heterogeneous population case studies

Heterogeneous populations that are not topologically similar pose different challenges when compared to both homogeneous populations and heterogeneous populations that are topologically similar. All three methods outlined in this paper require consistency in the dimensions of features between the source and target domain. From a structural dynamics perspective, the number of natural frequencies, damping ratios and modal vectors etc. would need to be equivalent in both domains. Another question therefore arises as to what level of consistency is required in the label space between the two domains. To discuss the required level of consistency, two case studies are presented. Both of these scenarios involve heterogeneous populations where each member is topologically dissimilar. The first case study considers a scenario where labels are considered consistent, and the second outlines an inconsistent label space problem.

These scenarios assess whether domain adaptation is applicable in the more general sense i.e. the heterogeneous case where structures in the populations are topologically dissimilar. This would allow labelled data from any structure to be mapped onto a different structure, given the condition that the feature vectors are the same dimension, meaning general classifiers could be constructed. This capability would mean that data from an experimental structure in a laboratory environment could be used to label data for any operational structure; providing significant benefits in the application of SHM across populations.

The two case studies in this section consider three and five-storey building structures. Both structures are constructed numerically as defined in Section 3.1, with the exception of the five-storey structure having an extra two mass, damping and stiffness terms i.e., $\{m_i, c_i, k_i\}_{i=1}^5$, located above the third storey. Again, damage is modelled as open cracks located in a single beam. The SHM problem for both scenarios is a two-class damage detection problem, where the source domain is labelled and the labels in the target domain are unknown. The features used in both examples are the first three bending, damped natural frequencies and damping ratios, meaning $X_s \in \mathbb{R}^{N_s \times 6}$ and $X_t \in \mathbb{R}^{N_t \times 6}$. Furthermore, a k-NN classifier is utilised in both studies, in order to highlight the effectiveness of TCA and JDA techniques where $k = 1$ and the distance is the Euclidean distance. ARTL was implemented using the regularised squared-loss hyperplane classifier approach.

4.1. Consistent labels

The first case study considers the scenario where damage labels are consistent between the source and target domains for a two-class damage detection problem. Specifically, this study considers that the labelled source domain data are generated from a five-storey building structure and the unlabelled target domain data are from a three-storey building structure. As stated, the first three bending, damped natural frequencies and damping ratios are utilised as features for both domains. The two classes in this problem are an undamaged case and a damaged case, where an open crack of 50% of the beam width is simulated at the midpoint of one beam located at the first degree-of-freedom, k_1, i.e. between the ground and the first storey. This is therefore a case where labels are consistent, i.e. the specific damage scenario can be found in both the five and three-storey structures.

The source and target domain structures both have the same geometric and material properties found in Table 5, with the structural differences occurring topologically, i.e. the number of storeys. 250 repeats for each class were obtained for the source domain; for the target domain 100 repeats for each class were used as training data and a further 250 repeats for each class used as testing data in the domain adaptation methods. Figs. 15 and 16 show the features for the source and target domains, where each quadrant is a two-dimensional comparison between feature combinations, ordered as natural frequencies followed by damping ratios, i.e. the bottom-left quadrant is the first natural frequency against the third damping ratio. The undamaged and damaged labels in these figures are '1' and '2' respectively.

To illustrate the applicability of domain adaptation in this context, a k-NN classifier was trained on the (normalised) source domain data and applied to the (normalised) target domain data. The accuracies and macro F_1-scores are both 0.167 and 33.3% respectively for the training and testing target domains, indicating a complete failure of the classifier to generalise.
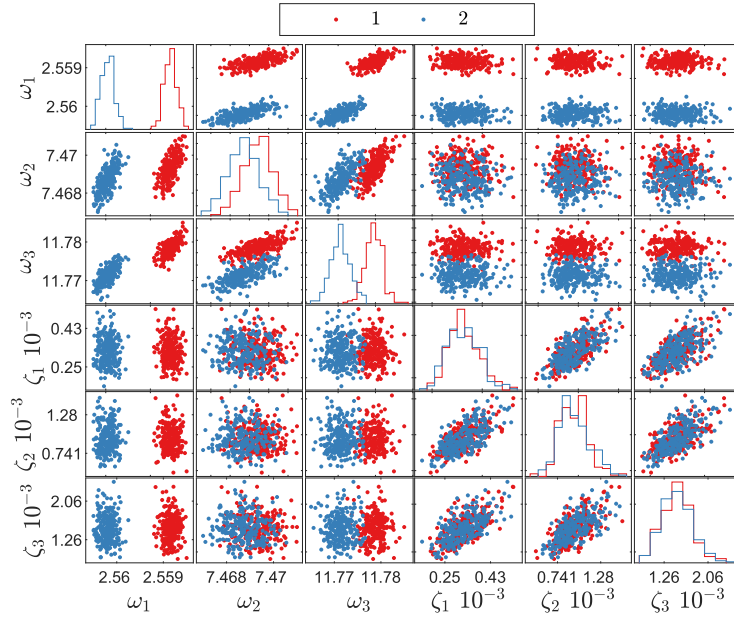
4.1.1. Domain adaptation

TCA, JDA and ARTL were implemented on the heterogeneous population with consistent labels. For each method, cross-validation was performed using a ten-fold approach such that the number of components and regularisation parameters

Table 5
Properties of the source and target structures for the consistent labels case study.

Property	Unit	Source	Target
Beam geometry, $\{l_b, w_b, t_b\}$	mm	$\{300, 40, 8\}$	$\{300, 40, 8\}$
Mass geometry, $\{l_m, w_m, t_m\}$	mm	$\{400, 400, 40\}$	$\{400, 400, 40\}$
Crack geometry, $\{l_{cr}, l_{loc}\}$	mm	$\{20.0, 150\}$	$\{20.0, 150\}$
Elastic modulus, E	GPa	$\mathcal{N}\left(210, 1 \times 10^{-10}\right)$	$\mathcal{N}\left(210, 1 \times 10^{-10}\right)$
Density, ρ	kg/m^3	$\mathcal{N}(7800, 10)$	$\mathcal{N}(7800, 10)$
Damping coefficient, c	Ns/m	$\mathcal{G}(8, 0.8)$	$\mathcal{G}(8, 0.8)$
Number of Storeys		5	3

Fig. 15. Source domain features for the consistent labels case study (where natural frequencies are in Hz).

could be determined. The macro F_1-score was the cost function for TCA and JDA, and the regularised squared loss implemented for ARTL. A linear kernel was used for each approach.

The number of components and regularisation parameter identified for TCA were two and $\mu = 1 \times 10^{-3}$. The transfer components are depicted in Fig. 17 and the marginal distributions shown in Fig. 18. These figures show that the first component satisfactorily maps the source and target domains onto the same space. In contrast, there is still distance between the second transfer component in the source and target domains. This happens to be beneficial as the target domain components are more separable, and is similar to what was observed in TCA and JDA in the multi-class case study. Here the target domain marginal distributions are approaching the source domain marginal distributions, and therefore are further away from the class boundary making the problem more separable.

JDA, on the other hand, satisfactorily (from a visual point of view) maps all the transfer components onto the same space as demonstrated in Fig. 19 and in the marginal and class-conditional distribution shown in Fig. 20. This indicates that the joint distributions are different for the source and target domains, as expected given that the features are from a heterogeneous population. However, the JDA transfer components for each label are closer together, meaning there is likely to be more confusion in any inferred classifier when compared to the TCA components. The number of components and regularisation parameters identified for JDA were five and $\mu = 1 \times 10^{-3}$ respectively.

ARTL cross-validation, using the regularised squared-loss cost function, identified one component and the following parameters: $\sigma = 1 \times 10^{-4}$, $\mu = 1$, $\gamma = 1 \times 10^{-4}$ and one nearest neighbour; ten iterations were used inside the algorithm.

Table 6 presents the classification results for the heterogeneous population with consistent labels case study; trained on the labelled source domain data and applied to the unlabelled target domain data sets. In this example, a k-NN classifier is trained on the transformed source domain components and tested on the training and testing target domain data for TCA and JDA. ARTL is implemented using the regularised squared-loss form. These findings show the benefits of domain adaptation for the heterogeneous population case. It is interesting to note that TCA outperforms JDA, even though as previously stated, JDA maps the source and target domains closer together. This difference in performance is likely due to the increased
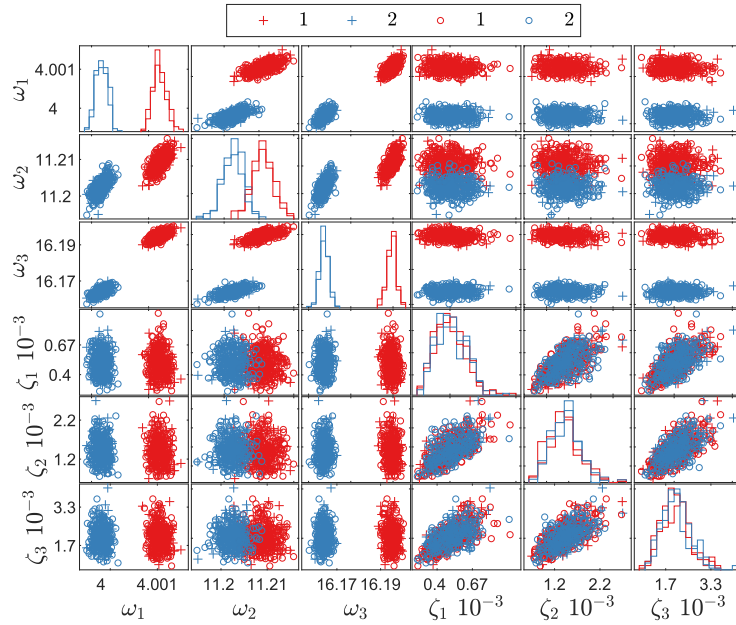
Fig. 16. Target domain features for the consistent labels case study (where natural frequencies are in Hz). The training and testing data are denoted by $(+)$ and $(\circ)$ respectively.
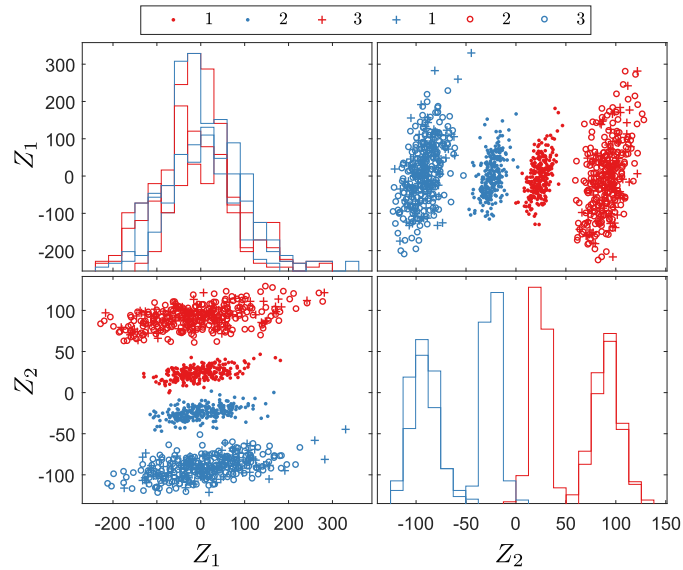

Fig. 17. A visualisation of the TCA-transformed components for the consistent labels case study. The source domain and the training and testing data for the target domain are denoted by $(\cdot), (+)$ and $(\circ)$ respectively.
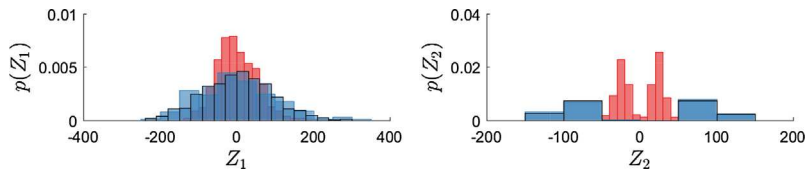

Fig. 18. The marginal distributions for each TCA-transformed component for the consistent labels case study. The source domain and target domain results are red and blue respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)
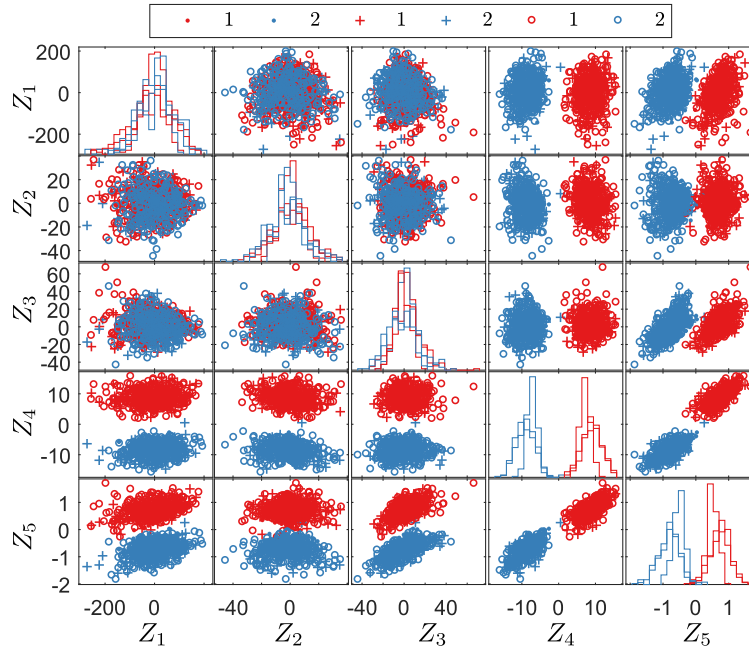
Fig. 19. A visualisation of the JDA-transformed components for the consistent labels case study. The source domain and the training and testing data for the target domain are denoted by $(\cdot), (+)$ and $(\circ)$ respectively.
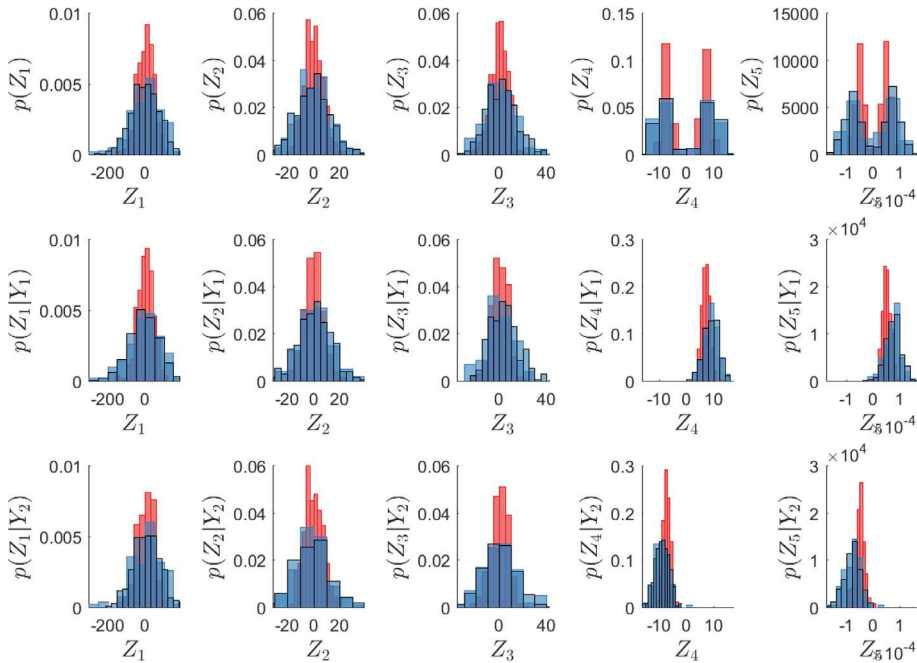

Fig. 20. The marginal and class-conditional distributions for each JDA-transformed component for the consistent labels case study. The source domain and target domain results are red and blue respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

separability in the TCA components as shown in Fig. 17. ARTL provides comparable results to TCA; this indicates that the realignment of the classification hyperplane onto the manifold can improve the results when reducing the distance between the joint distributions.

Table 6

Classification results for the consistent labels case study trained on the labelled source domain and applied to the unlabelled target domain.

Method		k-NN	TCA	JDA	ARTL
Mapping Training	Accuracy	50.0%	100.0%	91.0%	99.5%
	F_1-score	0.333	1.000	0.910	0.995
Mapping Testing	Accuracy	50.0%	100.0%	93.2%	100%
	F_1-score	0.333	1.000	0.932	1.000

4.2. Inconsistent labels

In order to demonstrate the applicability of domain adaptation on heterogeneous populations, where labelling is not consistent between the two structures, a final two-class damage detection case study is presented. The source and target domain data were obtained from a three and five-storey building structure respectively, where the properties are defined in Table 7.

In this scenario, damage occurred at the top degree-of-freedom for both structures, i.e. between the second and third floor for the three-storey building structure, k_3, and between the fourth and fifth floor for the five-storey building structure, k_5. This can be considered an inconsistent labelling scenario as the category of damage in the five-storey structure cannot occur in the three-storey structure as it does not have a fifth degree-of-freedom. The undamaged and damaged labels are denoted '1' and '2', where damage is simulated as a midpoint open crack at one of the beams at a particular degree-of-freedom. The features for this case study are presented in Fig. 21 and 22; these figures show that the source domain classes are more separable than the target domain. The number of data points for the source and target domains are $N_s = 500$ and $N_t = 200$, where data points are evenly split between classes; the number of data points used to test the target domain mapping was $N_{test} = 500$. A k-NN classifier trained on the (normalised) source domain data and applied to the (normalised) target domain data produced accuracies and macro F_1-scores of 50% and 0.333 for both the training and testing data. Again, this shows a need for domain adaptation, as the classifier fails to generalise between the source and target domains.

4.2.1. Domain adaptation

The three domain adaptation methods were utilised in generating classifiers that generalised between the source and target domains for a heterogeneous population with inconsistent labelling. Cross-validation, using ten-folds, was applied such that the regularisation parameters and number of components could be obtained. Linear kernels were also implemented.

TCA mappings are shown in Fig. 23, and the marginal distributions in Fig. 24. These figures display similar behaviour to the consistent label example, where the target domain component two is mapped in a more separable manner than in the source domain. JDA also produces similar results, as presented in Figs. 25 and 26; however, the marginal distributions for the second component are closer, likely due to the fact that $p(Y_s|\phi(X_s)) \neq p(Y_t|\phi(X_t))$. For both TCA and JDA, two components were identified via cross-validation with $\mu = 0.005$ and 50 respectively. When applied to ARTL with ten iterations, cross-validation using the regularised squared-loss cost function selected one transfer component, $\sigma = 1$, $\mu = 1$, $\gamma = 1$ and one nearest neighbour.

Table 8 presents a summary of the classification results for this case study; trained on the labelled source domain data and applied to the unlabelled target domain data sets. The table shows that all domain adaptation methods are applicable for a two-class damage detection problem in heterogeneous populations, where labelling is inconsistent and the features are consistent. This excellent classification performances are likely due to the fact that the labels can be considered consistent, in the sense that they both relate to 'damage' in a two-class detection problem, rather than being location specific.
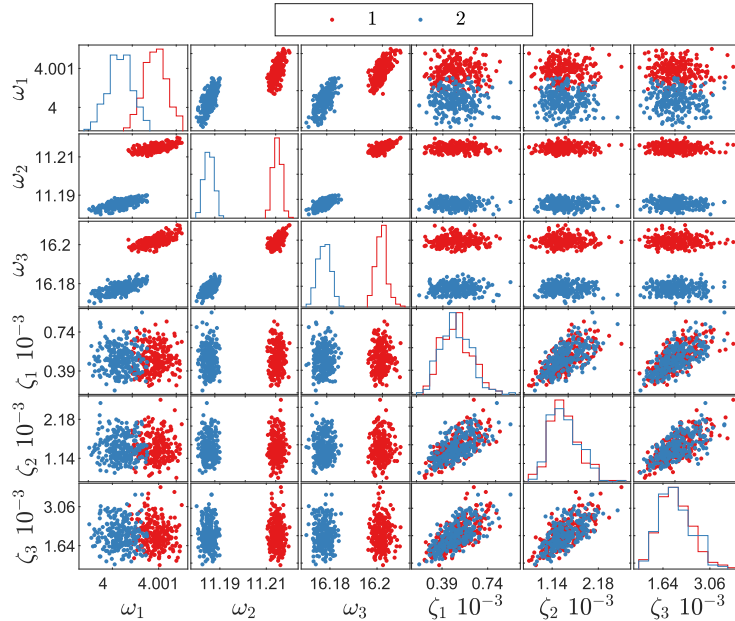
5. Conclusions

Domain adaptation has been demonstrated to be applicable in a variety of population-based SHM scenarios. This approach to transfer learning seeks to reduce the distance between data distributions in a source and target domain such that a classifier generalises between the two domains. As a result, supervised learning methods trained on a labelled source domain are applicable to an unlabelled target domain. In SHM, this situation occurs when labelled feature data are obtainable for one structure and classification is required on the unlabelled data of a different structure.

In discussing the applicability of domain adaptation for population-based SHM, four case studies were presented, categorised by whether the structures were heterogeneous populations with nominally-similar topologies or heterogeneous populations with differing topologies. Homogeneous populations, like a set of the same model wind turbine within a particular wind farm, are most likely to meet the mathematical requirements of domain adaptation, i.e. the feature set $\mathcal{X}_s = \mathcal{X}_t$ and labelled space $\mathcal{Y}_s = \mathcal{Y}_t$ are consistent. On the other hand, these requirements may also be met in certain heterogeneous populations, like those with nominally identical topologies, where for example transmissibilities over the same spectral lines are utilised as features, and labelling is consistent. However, if the more general case for heterogeneous populations, where consistency between these two objects is not met, then transfer learning offers a significant change in machine learning-based SHM. Currently, these algorithms rely on similarities between datasets in the source and target domains such that transfer learning can be performed. As structures and applications include more strongly heterogeneous populations, such as a

Table 7
Properties of the source and target structures for the inconsistent labels case study.

Property	Unit	Source	Target
Beam geometry, $\{l_b, w_b, t_b\}$	mm	$\{300, 40, 8\}$	$\{300, 40, 8\}$
Mass geometry, $\{l_m, w_m, t_m\}$	mm	$\{400, 400, 40\}$	$\{400, 400, 40\}$
Crack geometry, $\{l_{cr}, l_{loc}\}$	mm	$\{20.0, 150\}$	$\{20.0, 150\}$
Elastic modulus, E	GPa	$\mathcal{N}\left(210, 1 \times 10^{-10}\right)$	$\mathcal{N}\left(210, 1 \times 10^{-10}\right)$
Density, ρ	kg/m^3	$\mathcal{N}(7800, 10)$	$\mathcal{N}(7800, 10)$
Damping coefficient, c	Ns/m	$\mathcal{G}(8, 0.8)$	$\mathcal{G}(8, 0.8)$
Number of Storeys		3	5

Fig. 21. Source domain features for the inconsistent labels case study (where natural frequencies are in Hz).

population that includes rotating machinery with a cracked disc and a building structure with a crack on a floor, these mappings will need to be more complex, with the limitations of these approaches being an area for further research.

The specific domain adaptation algorithms investigated in this paper were TCA, JDA and ARTL, which differ in their progression of assumptions from marginal distribution adaptation, to joint distribution adaptation, and finally manifold regularisation with hyperplane inference. These methods all showed significant benefits in terms of classification performance, when compared against conventional supervised classification techniques. In addition, for each example, accuracies above 90% were produced for the majority of domain adaptation techniques.

Two case studies for heterogeneous populations that are topologically similar were presented, a multi-class case study and an application of domain adaptation from a numerical simulation as the source domain to an experimental target domain. The first case study performed damage detection and location between steel (source) and aluminium (target) three-storey building structures with different geometric dimensions. Classification results showed a progression of performance from TCA to ARTL. This means that the joint distributions were different between the two domains and that performance was improved by manifold regularisation. Furthermore, this work demonstrates that domain adaptation is applicable to any multi-class classification problem between topologically-equivalent structures, where labelled data is only known in the source domain. In this context, the problem of damage detection, location, inference of the type and extent are all possible, as long as labels are known in the source domain and the chosen features are separable in relation to their class labels.

The second topologically-equivalent case study presented the applicability of domain adaptation in resolving the lack of available damage-state data problem in SHM. In this scenario, a numerical simulation was utilised in providing labelled source domain data for an unlabelled experimental structure. TCA and JDA correctly classified all but one data point, and hence show that domain adaptation provides a solution to obtaining labelled feature data from physics-based models. This opens up a new category of approach in utilising physics-based models in SHM, and is highlighted as a significant area of further research within model-assisted SHM. It is noted that a better classifier than k-NN would still have had difficulties
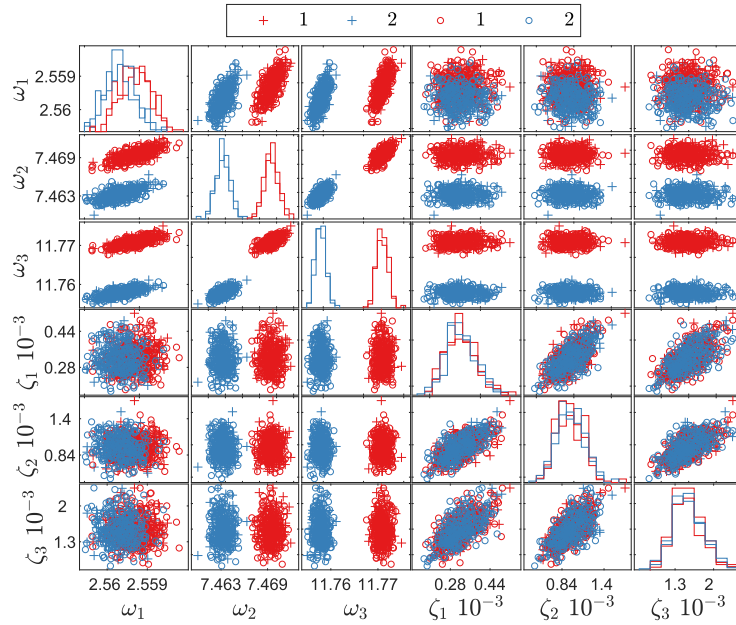
Fig. 22. Target domain features for the inconsistent labels case study (where natural frequencies are in Hz). The training and testing data are denoted by $(+)$ and $(\circ)$ respectively.
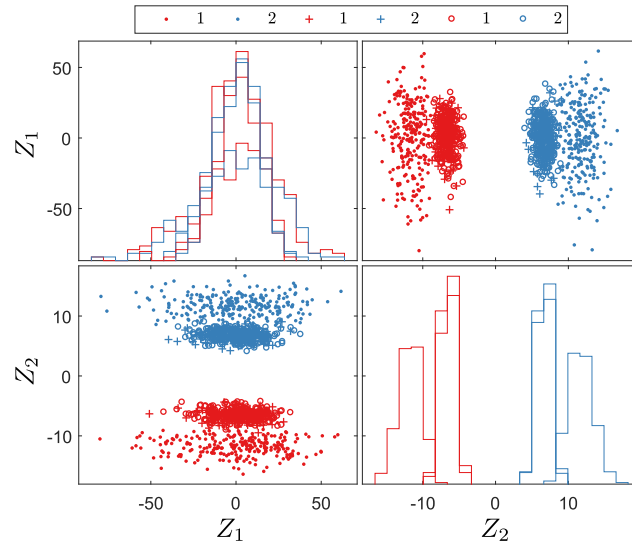

Fig. 23. A visualisation of the TCA-transformed components for the inconsistent labels case study. The source domain and the training and testing data for the target domain are denoted by $(\cdot), (+)$ and $(\circ)$ respectively.
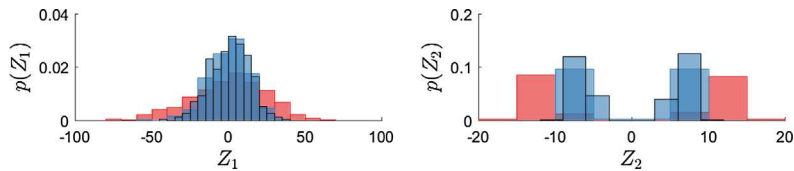

Fig. 24. The marginal distributions for each TCA-transformed component for the inconsistent labels case study. The source domain and target domain results are red and blue respectively.
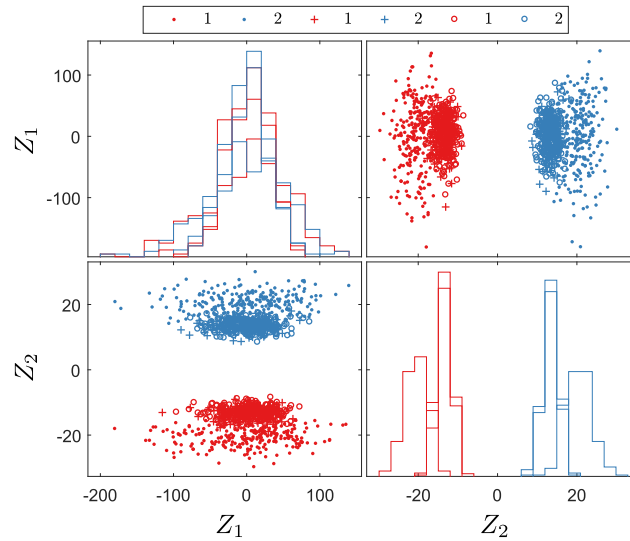
Fig. 25. A visualisation of the JDA-transformed components for the inconsistent labels case study. The source domain and the training and testing data for the target domain are denoted by $(\cdot), (+)$ and $(\circ)$ respectively.
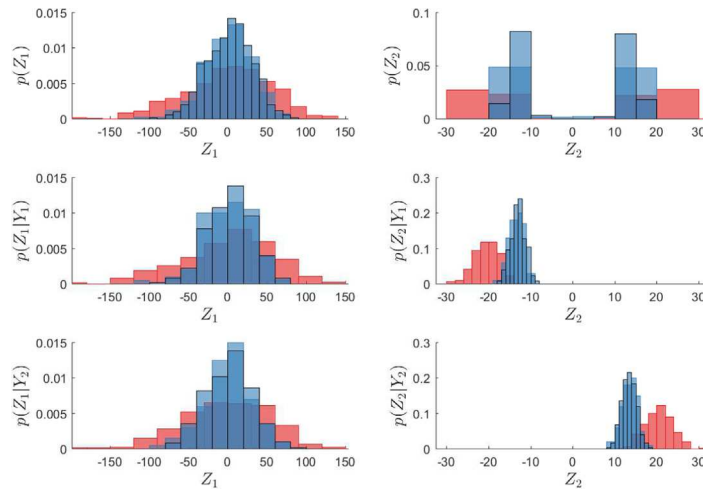

Fig. 26. The marginal and class-conditional distributions for each JDA-transformed component for the inconsistent labels case study. The source domain and target domain results are red and blue respectively. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 8
Classification results for the inconsistent labels case study trained on the labelled source domain and applied to the unlabelled target domain.

Method		k-NN	TCA	JDA	ARTL
Mapping Training	Accuracy	50.0%	100.0%	100.0%	100.0%
	F_1-score	0.333	1.000	1.000	1.000
Mapping Testing	Accuracy	50.0%	100.0%	100.0%	100.0%
	F_1-score	0.333	1.000	1.000	1.000

in classifying all the of the target domain correctly for both TCA and JDA. This is because any bound inferred from the source domain would not have had enough information to correctly classify the single misidentified data point.

In terms of heterogeneous populations with dissimilar topologies, two scenarios were considered: consistent and inconsistent feature labelling. In the first case study, the damage scenario was the same between a five and three-storey building structure, namely an open crack in a beam at the first degree-of-freedom. TCA and ARTL both produced 100% classification

accuracies on test data in the target domain. This result demonstrates that if the features and labels are consistent then domain adaptation can be applied to heterogeneous populations with non-identical topologies.

The final case study involved inconsistent labelling between a three (source) and five (target) storey building structure. This breaks the assumption contained within domain adaptation, that $\mathcal{Y}_s = \mathcal{Y}_t$. However, in this scenario, where an open crack was considered at a beam between the top two floors, macro F_1-scores of 1.000 were produced for all three methods. This is likely due to increased separability in the source domain features, meaning that the target domain is made more separable in the transformed space. However, applying domain adaptation in this context is not guaranteed to produce good results as it invalidates the label space assumption. In this example, the methods are likely to have worked as it was a two-class detection problem where the label can be considered 'damaged' and does not relate to the location, type or extent.

Finally, this paper has outlined several novel applications of domain adaptation within population-based SHM. TCA, JDA and ARTL — all new techniques within SHM — have shown significant increases in classification performance, where generally ARTL performs best when the conditional distributions are not equal between the source and target domain; otherwise, TCA provides satisfactory performance. Further research should be conducted in developing techniques for the general heterogeneous population case, i.e. where the feature and label space are inconsistent. In addition, research should be conducted in utilising better semi-supervised labelled methods with JDA and ARTL rather than a naive pseudo-labelling approach.

CRediT authorship contribution statement

P. Gardner: Conceptualization, Formal analysis, Investigation, Software. **X. Liu:** Conceptualization, Investigation. **K. Worden:** Conceptualization, Funding acquisition, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

References

[1] K. Worden, J.M. Dulieu-Barton, An overview of intelligent fault detection in systems and structures, Struct. Health Monit. 3 (1) (2004) 85–98.
[2] K. Worden, G. Manson, The application of machine learning to structural health monitoring, Philos. Trans. R. Soc. A: Math., Phys. Eng. Sci. 365 (1851) (2007) 515–537.
[3] E. Figueiredo, G. Park, C.R. Farrar, K. Worden, J. Figueiras, Machine learning algorithms for damage detection under operational and environmental variability, Struct. Health Monitor. 10 (6) (2011) 559–572.
[4] C.R. Farrar, K. Worden, Structural Health Monitoring: A Machine Learning Perspective, John Wiley & Sons Ltd, Chichester, UK, 2012.
[5] E. Papatheou, N. Dervilis, E.A. Maguire, I. Antoniadou, K. Worden, Population-based SHM: a case study on an offshore wind farm, in: 10th International Workshop on Structural Health Monitoring, 2015.
[6] A. Rytter, Vibrational based inspection of civil engineering structures (Ph.D. thesis), 1993..
[7] S.J. Pan, Q. Yang, A survey on transfer learning, IEEE Trans. Knowl. Data Eng. 22 (10) (2010) 1345–1359.
[8] S. Dorafshan, R.J. Thomas, M. Maguire, Comparison of deep convolutional neural networks and edge detectors for image-based crack detection in concrete, Constr. Build. Mater. 186 (2018) 1031–1045.
[9] Y. Gao, K.M. Mosalam, Deep transfer learning for image-based structural damage recognition, Comput.-Aided Civil Infrastruct. Eng. 33 (9) (2018) 748–768.
[10] K. Jang, N. Kim, Y.-K. An, Deep learning-based autonomous concrete crack evaluation through hybrid image scanning. Struct. Health Monitor., 2019, page 147592171882171..
[11] D. Chakraborty, N. Kovvali, B. Chakraborty, A. Papandreou-Suppappola, A. Chattopadhyay, Structural damage detection with insufficient data using transfer learning techniques, in: Sensors and Smart Structures Technologies for Civil, Mechanical, and Aerospace Systems, 2011, p. 798147.
[12] J. Ye, T. Kobayashi, H. Tsuda, M. Murakawa, Robust hammering echo analysis for concrete assessment with transfer learning, in: Proceedings of the 11th International Workshop on Structural Health Monitoring, 2017, pp. 943–949..
[13] M. Sugiyama, S. Nakajima, H. Kashima, P. Von, M. Kawanabe, Direct importance estimation with model selection and its application to covariate shift adaptation, in: Advances in Neural Information Processing Systems, 2008..
[14] B. Quanz, J. Huan, M. Mishra, Knowledge transfer with low-quality data: a feature extraction issue, IEEE Trans. Knowl. Data Eng. 24 (10) (2012) 1789–1802.
[15] M. Baktashmotlagh, M.T. Harandi, B.C. Lovell, M. Salzmann, Domain adaptation on the statistical manifold, in: IEEE Conference on Computer Vision and Pattern Recognition, 2014, pp. 2481–2488.
[16] K.M. Borgwardt, A. Gretton, M.J. Rasch, H.-P. Kriegel, B. Scholkopf, A.J. Smola, Integrating structured biological data by kernel maximum mean discrepancy, Bioinformatics 22 (14) (2006) 49–57.
[17] S.J. Pan, I.W. Tsang, J.T. Kwok, Q. Yang, Domain adaptation via transfer component analysis, IEEE Trans. Neural Networks 22 (2) (2011) 199–210.
[18] B. Schölkopf, A. Smola, K.-R. Müller, Nonlinear component analysis as a kernel eigenvalue problem, Neural Comput. 10 (5) (1998) 1299–1319.
[19] M. Long, J. Wang, G. Ding, J. Sun, P.S. Yu, Transfer feature learning with joint distribution adaptation, in: IEEE International Conference on Computer Vision, 2013, pp. 2200–2207.
[20] M. Long, J. Wang, G. Ding, S.J. Pan, P.S. Yu, Adaptation regularization: a general framework for transfer learning, IEEE Trans. Knowl. Data Eng. 26 (5) (2014) 1076–1089.
[21] B. Schölkopf, R. Herbrich, A.J. Smola, A generalized representer theorem, in: Computation Learning Theory, Springer, 2001, pp. 416–426.

[22] M. Belkin, P. Niyogi, V. Sindhwani, Manifold regularization: a geometric framework for learning from examples, J. Mach. Learn. Res. 7 (2006) 2399–2434.
[23] S. Christides, A. Barr, One-dimensional theory of cracked Bernoulli-Euler beams, Int. J. Mech. Sci. 26 (11–12) (1984) 639–648.
[24] M.I. Friswell, Damage identification using inverse methods, Philos. Trans. R. Soc. A: Math., Phys. Eng. Sci. 365 (1851) (2007) 393–410.