Chapter 14: Machine Learning and Pattern Recognition: Methods and Applications for Integrity Monitoring of Civil Engineering Structures

Chapter details

Chapter DOI:

https://doi.org/10.4322/978-65-86503-71-5.c14

Chapter suggested citation / reference style:

Alves, Vinicius N., et al. (2022). "Machine Learning and Pattern Recognition: Methods and Applications for Integrity Monitoring of Civil Engineering Structures". *In* Jorge, Ariosto B., et al. (Eds.) *Model-Based and Signal-Based Inverse Methods*, Vol. I, UnB, Brasilia, DF, Brazil, pp. 502–535. Book series in Discrete Models, Inverse Methods, & Uncertainty Modeling in Structural Integrity.

P.S.: DOI may be included at the end of citation, for completeness.

Book details

Book: Model-based and Signal-Based Inverse Methods

Edited by: Jorge, Ariosto B., Anflor, Carla T. M., Gomes, Guilherme F., & Carneiro, Sergio H. S.

Volume I of Book Series in:

Discrete Models, Inverse Methods, & Uncertainty Modeling in Structural Integrity

Published by: UnB City: Brasilia, DF, Brazil Year: 2022

DOI: https://doi.org/10.4322/978-65-86503-71-5

Machine Learning and Pattern Recognition: Methods and Applications for Integrity Monitoring of Civil Engineering Structures

Vinicius N. Alves[1a], Alexandre A. Cury[*2], Ney Roitman[3a], Carlos Magluta[3b]

[1]Graduate Program in Civil Engineering, Federal University of Ouro Preto, Brazil. E-mail: vinicius.alves@ufop.edu.br
[2]Graduate Program in Civil Engineering, University of Juiz de Fora, Juiz de Fora, Brazil. E-mail: alexandre.cury@ufjf.edu.br
[3]Federal University of Rio de Janeiro, Civil Engineering Department, COPPE, Rio de Janeiro, Brazil. E-mail: roitman@coc.ufrj.br, magluta@coc.ufrj.br

*Corresponding author: alexandre.cury@ufjf.edu.br

Abstract

Structural Health Monitoring (SHM) is a problem that can be addressed at many levels. One of the most promising approaches used in damage assessment problems is based on pattern recognition. The idea is to extract features from data that characterize only the normal condition and to use them as a template or reference. During structural monitoring, several data are measured, and appropriate features need to be extracted and compared to a baseline. Any significant deviations could be considered as structural novelty or possible damage. This chapter presents a collection of novelty detection approaches where the concept of Symbolic Data Analysis (SDA) is used to manipulate raw vibration data (i.e., acceleration measurements). These quantities (transformed into symbolic data) are combined to unsupervised - hierarchy-agglomerative, dynamic clouds, and soft c-means clustering – and supervised classification techniques - Bayesian decision trees, artificial neural networks, and support vector machines applied to SHM. To attest the robustness of these approaches, experimental tests are performed on a simply supported beam considering different damage scenarios, and on a motorway bridge, in France, where thermal variation effects also played a major role. The results obtained confirm the efficiency of the proposed methodologies. Finally, the authors present some practical recommendations about the discussed techniques.

1. Introduction

Studies related to early damage detection are of special concern for civil engineering structures. It is common knowledge that if a damage process is not identified in time, structural systems may undergo serious safety and economic consequences. Traditional methods of damage detection and health monitoring are often based on the variation of structural vibration characteristics, i.e., natural frequencies, damping ratios and mode shapes. Any modification of mechanical properties must be detectable through changes in the modal parameters (Moughty and Casas [2017]). Research in vibration-based damage identification has been rapidly expanding over the last few years. In their survey, Doebling et al. [1996] have addressed several critical issues for future research in damage identification and health monitoring. Although this literature survey is now almost 30 years old, most of the conclusions are still valid despite the large research work done since then.

As previously mentioned, most of the techniques used for damage identification are essentially based on the determination of modal properties through an identification process (Cury et al. [2011]; Hakim and Razak [2014]). Nevertheless, the identification of modal parameters is a sort of filtering process, leading to a loss of information compared to raw data (acceleration measurements). This compression process can erase any small changes due to a structural modification. Furthermore, and this is certainly the major drawback when using modal parameters, modal components are essentially describing an equivalent linear behavior, a feature which may be not exact for the analysis of specific degraded systems (Finotti et al. [2019]). In turn, using raw dynamic measurements (especially if high sampling frequencies are used) leads to the storage of large data sets. Dynamic measurements can easily contain over thousands of values, making an analysis process extensive and prohibitive. Few damage detection methods present in the literature are based on signature principles, but they usually fail when making them practical. In this sense, despite the current processing power of computers, the necessary computational effort to manipulate large datasets remains a problem. A lot of effort has been put in the development of new procedures of damage detection involving not only engineers, but also researchers of different areas such as mathematics, physics, statistics, among others, to improve formulations of well-known techniques or to develop new ones (Cardoso et al. [2019]; Santos et al. [2020]; Zhou et al. [2019]).

The arising problem often faced by structural engineering when it comes to health monitoring is the great amount of data acquired through dynamic tests. More important than gather this information is, of course, the interpretation of these data. It is difficult, however, to analyze this type of (raw) information although extremely important in some cases where the engineer/stakeholder needs to know, in real time, the condition of a given structure. Thus, it would be imperative to assess structural information directly from raw data i.e., accelerations measured *in situ*.

The development of high-performance sensors, precision signal conditioning, analog-to-digital converters, optical or wireless networks, global positioning systems and so on, has drastically changed the vision of structural monitoring, giving engineers a large amount of data and consequently performance indicators. In connection with advanced software for a structural analysis, significant developments can be expected regarding the detection of deterioration mechanisms. These developments have opened the way for a wide range of applications dedicated to efficient operation and maintenance of civil engineering structures.

Detecting structural changes in a timely manner and understanding the mechanical behavior are critical to ensure that the resulting disruption and the economic management issues are optimized. This explains why many expectations have been placed in vibration-based monitoring for structural behavior characterization and novelty detection (diagnosis of abnormal behavior) (Cury and Crémona [2012]). Many novelty techniques have been proposed in the last few years to detect structural damage. The use of most of these techniques yielded interesting and encouraging results. In fact, in some applications, it was possible to identify several structural conditions correctly i.e., to discriminate undamaged conditions from damaged behaviors using modal parameters (natural frequencies, mostly) (Alvandi and Cremona [2006]; Wang [2013]). However, the use of these techniques applied to raw data (accelerations) remains a challenge. To overcome these limitations, this chapter presents the use of a special set of data manipulation techniques. Data mining is the process of extracting hidden patterns or features from data. As more data are gathered in monitoring, data mining is becoming an increasingly important tool to transform these data into information and is being used in a wide range of profiling practices, such as marketing, fraud detection and scientific discovery.

Different types of data can be employed and manipulated in data mining, such as single quantitative or categorical values, interval-valued data, multi-valued categorical data, and modal multi-valued (histograms) (Cury and Crémona [2012]). These types of data are generally called “symbolic data” and they allow representing the variability and uncertainty present in each variable. The development of new methods for data analysis suitable for treating this type of data is the main issue of Symbolic Data Analysis (SDA). This chapter shows how the combination of SDA with classification methods can be used to separate different structural states. The major advantage of such combined approach is that enhanced - yet raw - information is used, i.e., histograms, intervals, etc. and they can be applied to manipulate vibration data (acceleration measurements, for example). When these quantities are converted into symbolic data, this piece of information will be applied to three clustering methods: hierarchy-agglomerative, dynamic clouds and soft c-means clustering and three supervised classification techniques: Bayesian decision trees, artificial neural networks, and support vector machines. The main objective here is to assess structural modifications due to damage or any other abnormal event, such as reinforcement procedures, different types of traffic loads, among others. However, when applying vibration-based damage detection to SHM, changes in vibration signatures are

not only based on changes in any physical property. Environmental changes, notably temperature variations, can have a significant effect and it is necessary to take this into account for the evaluation of structural integrity (Martins et al. [2014]; Xia et al. [2013]). Encouraging results are obtained and they show strong evidence that environmental effects play an important role in the field of SHM.

This chapter is organized as follows: Section 1 presents a brief state-of-art about SHM techniques in the framework of civil engineering structures. Section 2 delves into some thoughts on machine learning techniques. Initially, it explains the idea behind the concept of Symbolic Data Analysis. Then, it presents a contextualized explanation about unsupervised and supervised classification methods applied to structural novelty detection. Section 3 focuses on the applications and the results obtained using the proposed approach. Finally, Section 4 presents the final remarks and some recommendations about the practical use of the techniques discussed in this chapter.

2. Machine Learning Techniques Overview

This section intends to give the reader a general perspective about how vibration data can be compressed while keeping their intrinsic characteristics necessary to provide enough information about the structure's dynamic behavior. Next, it explains how these condensed data is inputted to classification techniques, which ultimately will discriminate abnormal structural conditions.

2.1 Symbolic Data Analysis

In general, data acquisition campaigns in civil engineering structures gather thousands of accelerations values measured by several sensors. Consequently, analyzing all these data (classical data) directly may usually be time-consuming or even prohibitive. In this sense, transforming this massive quantity of data into a compact but also rich descriptive type of data (symbolic data) becomes an attractive approach. Let us consider, for instance, a signal X (which is part of a dynamic test) containing 5,000 acceleration values measured by one single sensor (see Fig.1 on the left). There are several ways to transform classical data into symbolic data.

This signal can be represented by:

- a *k*-category histogram: X={1(0.0025), 2(0.0721), 3(0.8546), 4(0.0626), …, k(0.0082)};
- an interquartile interval: X = [-0.012; 0.015];
- a min/max interval: X = [-0.025; 0.025].

Figure 1 (on the right) shows how a classical signal (one sensor) is converted to a symbolic representation. In this case, all acceleration values are projected to the y-axis of coordinates and a 20-category histogram is constructed. In fact, it must be noted that the same representation could be applied to modal parameters, i.e., natural frequencies and

mode shapes. In other words, both quantities can be represented by intervals or histograms. Transforming classical data to symbolic data is carried out almost instantaneously, which does not prohibit or make difficult the use of this methodology for a large ensemble of dynamic tests.

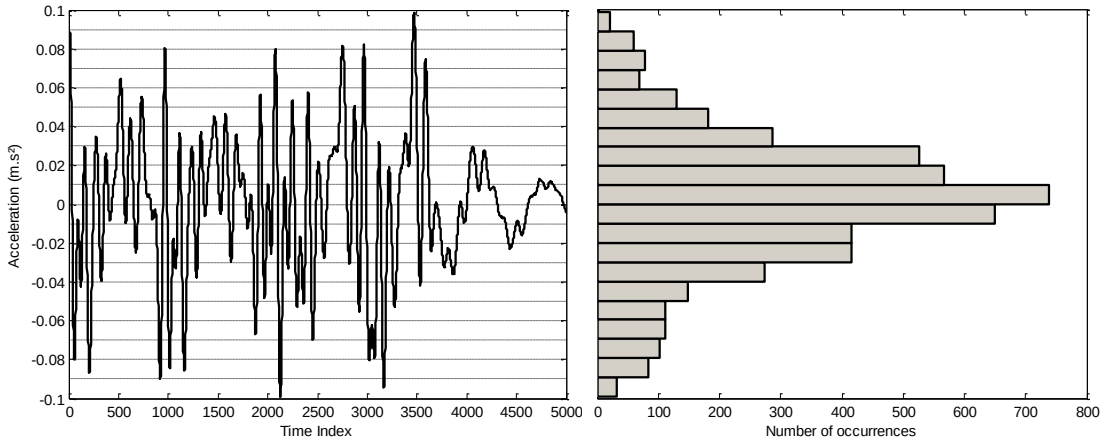

Figure 1 – Example of transforming classical signal to symbolic signal (20-category histogram).

In fact, when this transformation procedure is carried out, two important aspects must be considered. The first one relies on the conservation of some statistical properties of the original data, i.e., the moments of first order (mean value) and second order (variance). Higher order moments (skewness and kurtosis) are not considered here. The second aspect refers to the number of nonzero categories. It is not of interest to keep categories with values equal to zero since they will not contribute to the classification procedures. Thus, in this chapter, 10-category histograms are used in the SDA process, since it has proven to be the most efficient transformation for this type of analysis (Alves et al. [2015].

2.2 Unsupervised classification methods

Data clustering is a common technique for statistical data analysis, which is used in many fields, including machine learning, data mining, pattern recognition, image analysis and bioinformatics (Madhulatha [2012]). A clustering procedure can be defined as a way of classifying several objects into different groups. More precisely, it can be described as the partitioning of a data set into subsets (clusters), so that the data in each subset share some common properties. For an appropriate clustering, it is necessary to minimize the within-cluster variation to obtain the most homogeneous clusters as possible and, as a natural consequence, to maximize the between-cluster variation to obtain the most dissimilar clusters among each other. To define these clusters and determine the proximity (or similarity) among tests, it is necessary to define suitable dissimilarity measures. In a common sense, the lower these values are the more similar the objects are and thus, they are gathered in the same cluster. Conversely, the objects allocated into different clusters are the ones that have greater distances between them. Dissimilarity

measures can take a variety of forms and some applications might require specific ones. More details can be found in (Billard and Diday [2006]).

In this chapter, three clustering methods are used to discriminate structural conditions: hierarchy-agglomerative, dynamic clouds and fuzzy c-means. The first two are briefly described here since reference (Finotti et al. [2019]) contains additional details. Thus, their formulations are omitted in this chapter. The third method, since it is less documented in this field of research, is described later.

2.2.1 Hierarchy-Agglomerative

Hierarchy-Agglomerative is a bottom-up clustering process, i.e. the process starts with q clusters containing one single test, and proceeds by merging two sub-clusters (C^1 and C^2, say) into one new cluster C. The sub-clusters are merged according to similar criteria for minimizing the within-cluster variation and for maximizing the between-cluster variation. More details can be found in (Billard and Diday [2006]; Finotti et al. [2019]). This clustering method provides a measure of proximity between clusters when using the "difference in height" between them. Fig. 2 illustrates a hierarchy-agglomerative clustering procedure. In this example, clusters 1, 2 and 3 are highlighted. The heights "H(1,2)" and "H(2,3)" represent the distances among them. As "H(1,2)" is greater than "H(2,3)", clusters 2 and 3 are closer (or more similar) than clusters 1 and 2. For this method, the categorical symbolic distance (Billard and Diday [2006]) and the centroid linkage clustering were adopted. These parameters were chosen after adequate results obtained by (Alves et al. [2015]).

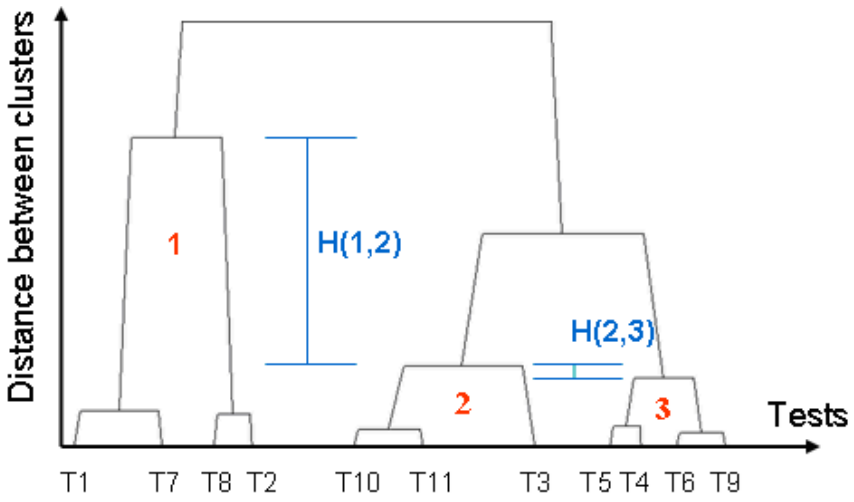

Figure 2 – Example of a hierarchy-agglomerative clustering.

2.2.2 Dynamic Clouds

This clustering method is based on a generalization of the classical dynamical clusters' method (Billard and Diday [2006]). This method consists in minimizing a general optimized criterion that measures the adequacy between the partition and the representation of the clusters, denoted *prototype*. A prototype is a symbolic description

model representing a cluster or, in other words, the “average” concept of a cluster. It is used as a reference to calculate the distances among the concepts and to define each cluster.

The algorithm initiates with a set of k random prototypes and iteratively applies an allocation phase to place each concept in the cluster where the proximity between concept and prototype is minimal. In this process, a representation phase is performed where the prototypes are updated according to the allocation phase results. This is realized by computing and storing the total sum of the distances between concepts and the prototype in one cluster. The new cluster prototype is the one minimizing this sum. These two phases procedure is repeated until convergence, that is, when the adequacy criterion reaches a stationary value (or at least when reaching a maximum number of iterations).

In general, the dynamic cluster algorithm converges in a few iterations. To improve the quality of the clustering, the algorithm is executed with different initial partitions, and the best configuration is chosen among all the results. Fig. 3 shows a simplified scheme of this clustering method. For this method, the categorical symbolic distance (Billard and Diday [2006]) was adopted. These parameters were chosen after adequate results obtained by (Alves et al. [2015]).

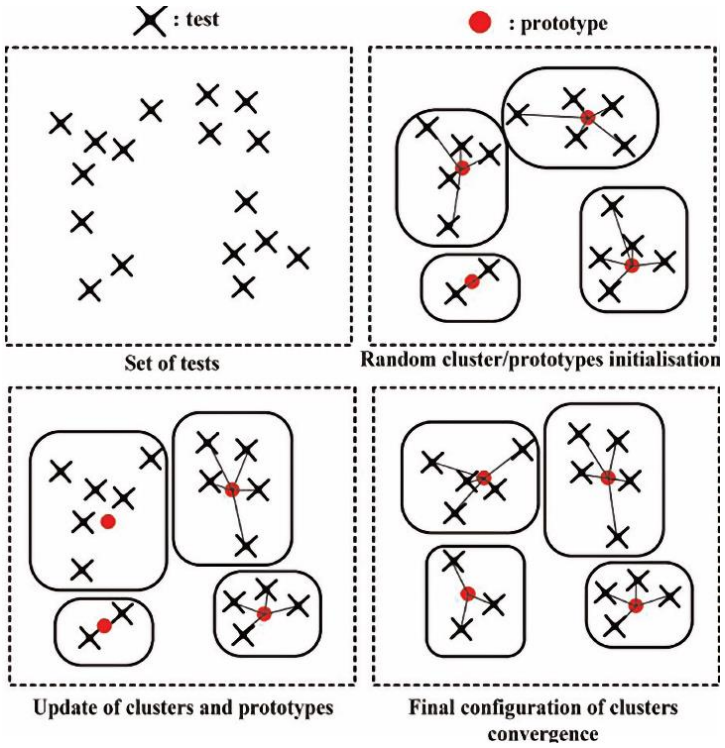

Figure 3 – Scheme representing the dynamic clouds algorithm.

2.2.3 Fuzzy c-means method (FCM)

A fuzzy set is a class of objects that has a continuum of grades of membership. Such a set is characterized by a membership (characteristic) function that assigns a grade of membership that ranges between zero and one to each object (Kroszynski and Zhou

[1998]; Song et al. [2006]). The FCM is an iterative algorithm clustering method that produces optimal c-partitions by minimizing the weighted dissimilarity function (Bezdek [1981]).

Let T_i $(i = 1,,2,...,n)$, which represents a *m*-dimensional vector (*m* is the number of categories of each dynamic test transformed into symbolic data). This notation is used to determine the cluster centers CC_{qr} for the q^{th} cluster and its r^{th} dimension by using the expression given below:

$$CC_{qr} = \frac{\sum_{i=1}^{n} U_{iqr}^{f} T_{ir}}{\sum_{i=1}^{n} U_{iqr}^{f}} \tag{1}$$

where C is the number of the cluster to be made $(2 \le C \le N)$, and f is an appropriate level of cluster fuzziness $f > 1$. U is the membership matrix, which has the size $n \times C \times m$ and is first initialized randomly such that $U_{iqr} \in [0,1]$ and $\sum_{q=1}^{C} U_{iqr} = 1$, for each i and a fixed value of *r*.

Then, the Euclidean distance is evaluated between the i^{th} data point and the q^{th} cluster with respect to the r^{th} dimension with the equation below:

$$D_{iqr} = \left\| T_{ir} - CC_{qr} \right\| \tag{2}$$

In the following step, CC_{qr} is updated in Eq.(1) by recalculating the fuzzy membership matrix U according to D_{iqr} (if $D_{iqr} > 0$), as follows:

$$U_{iqr} = \frac{1}{\sum_{c=1}^{C} \left(\frac{D_{ir}}{D_{icr}} \right)^{f^{2}-1}} \tag{3}$$

This updating iteration is repeated until the changes in U are sufficiently small (such that $(U \le \varepsilon)$, where ε is a predefined termination criterion.

The main difference between hard (hierarchy agglomerative and dynamic clouds) and soft-clustering (FCM) relies on the fact that hard-clustering methods can only assign a given object to a unique cluster. Conversely, soft-clustering methods allow classifying a given object as a part of different the clusters. This allows defining the so-called *pertinence values.* The higher these values are, more likely a given object is to be classified into a cluster. In this chapter, the aforementioned procedure was coupled with symbolic data transformation within Matlab® software framework. In fact, fuzzy c-means procedures are built-in in the Fuzzy Toolbox. The Euclidian distance was used, and the level of fuzziness was set to 2.

2.2.4 Determination of the optimal number of clusters

A common limitation of the clustering methods cited before is the necessity of predefining the number of clusters. In practice, however, this number is usually unknown. To circumvent this drawback, many different stopping rules for determining the optimal number of clusters have been published in the scientific literature. The most detailed and complete comparative study has been carried out by (Milligan and Cooper [1985]). They performed a Monte Carlo evaluation of thirty indexes for the determination of the optimal number of clusters and they investigated the extent to which these indexes were able to detect the correct number of clusters in a series of simulated data sets containing a known structure. In this section, the three best indexes are presented: the CH index, the C^* index and the Γ index (Billard and Diday [2006]).

The general methodology for the evaluation of these three indexes is based on calculating each one of them for a partition $P_q = (C^1, \cdots, C^q)$ containing a different number of clusters. The number of clusters q is arbitrarily chosen, but it is usually greater than the number of clusters considered in the analysis. Since $q = 1$ represents the partition P_1, which only contains the initial cluster, the indexes are not considered in this case.

The CH index is given by:

$$CH(P_j) = \frac{B(P_j)}{W(P_j)} \times \frac{(n-j)}{(j-1)}, \; j = 2,...,q \tag{4}$$

where $B(P_j)$ is the between-cluster variation, $W(P_j)$ is the total within-cluster variation, n is the total number of tests and j is the number of clusters in the partition P_j. Further details can be found in (Finotti et al. [2019]). When CH has its **maximal absolute value**, then the optimal partition (i.e., the optimal number of clusters) is obtained.

The C^* index can be calculated as:

$$C^*(P_j) = \frac{1}{n} \sum_{k=1}^{j} n_k \frac{(S^k - S^k_{\min})}{(S^k_{\max} - S^k_{\min})}, \; j = 2,...,q \quad C^* \in [0,1] \tag{5}$$

where n is the total number of tests, n_k is the number of tests of a cluster C^k, S^k represents the sum of distances among the k tests within a cluster C^k, $S^k_{\min}$ is the sum of the k smallest distances among all tests, $S^k_{\max}$ is the sum of the k largest distances among all tests. The optimal partition is given for C^* **minimal absolute value**.

The Γ index is given by:

$$\Gamma(P_j) = \frac{\Gamma_+(P_j) - \Gamma_-(P_j)}{\Gamma_+(P_j) + \Gamma_-(P_j)},\ j = 2,...,q \qquad \Gamma \leq 1 \tag{6}$$

where $\Gamma_+(P_j)$ represents the number of within-cluster distances smaller than the between-cluster distances and $\Gamma_-(P_j)$ is the number of within-cluster distances larger than the between-cluster distances. The optimal partition is given for **4 maximal absolute value**.

2.3 Supervised classification methods

This section presents an overview of three supervised classification methods. Firstly, some concepts regarding Bayesian Decision Trees (BDT) and its applicability are explained, followed by a brief discussion about how Neural Networks (NN) are used in this study. Finally, a general idea of Support Vector Machines (SVM) applied to classification problems is presented. These methods were selected following the works of Martins et al. [2014] and Xia et al. [2013]. In those references, several supervised classification techniques coupled with SDA were tested using either artificial or real controlled (labeled) data. In general, BDT, NN and SVM achieved the best correct classification ratios.

2.3.1 Bayesian Decision Trees

Bayesian Decision Trees (BDT) are a decision procedure that can solve classification problems (Billard and Diday [2006]). The general idea of this method is to classify a particular object (dynamic test, in this case) into one of the classes (groups) previously defined i.e., in the training set. For instance, let $\Omega = \{T_1, T_2, ..., T_n\}$ be a set of *n* dynamic tests and *C* a class variable containing values varying within $\{1,...,m\}$ where *m* is the number of classes. This discriminant analysis tries to predict the unknown value *C* for a given test $\tilde{T}$ according to its *p* features (the symbolic representations of sensors) and a training set. The steps of this symbolic classification procedure are to represent a given partition in the form of a BDT and create a rule capable of assigning a new test to one class of a prior partition.

Now, let us consider that each test T_i is described by two types of variables, considering symbolic signals:

- *p* sensors described as *k*-category histograms;
- a class variable *C*: this variable specifies the class of a test in the training set in the form of a unique value (1, 2,…, *m*).

Let T_1 denote a test belonging to class 1 and T_2 a test belonging to class 2 within the training set. The goal is to classify a test $\tilde{T}$ into one of these two classes. In this example, all tests are described by two sensors only (s_1, s_2).

In the framework of the recursive algorithm, each node division step is performed according to a single variable (suitably chosen) and to "yes/no" answers to specific binary

questions. The set of binary questions that will be useful for the recursive partition procedure is based on the Bayesian rule (Billard and Diday [2006]). This rule can be enounced as follows: suppose that for a set of tests Ω, the test $\tilde{T}$ is distributed with density functions:

$$f_j(\tilde{T}), \quad j=1,...,m \tag{7}$$

Density functions are generally unknown and need to be estimated. One way to solve this issue is to use the Kernel method (Kroszynski and Zhou [1998]), which can reasonably approximate density functions. The Kernel estimator is defined as:

$$f_j(\tilde{T}) \cong \frac{1}{n_j}\sum_{s=1}^{n_j}\sum_{r=1}^{p}\sum_{q=1}^{k}\frac{1}{h}K\left(\frac{\tilde{T}^{r,q}-T_s^{r,q}}{h}\right), \; j=1,...,m \tag{8}$$

where n_j is the number of tests within the class *j*, $h>0$ is a smoothing parameter and *K* is called kernel, which is symmetric, continuous and can be evaluated by:

$$K\left(\frac{\tilde{T}^{r,q}-T_s^{r,q}}{h}\right)=\frac{1}{\sqrt{2\pi}}e^{-\frac{(\tilde{T}^{r,q}-T_s^{r,q})^2}{2h^2}}, s=1,...,n_j; r=1,...,p; q=1,...,k \tag{9}$$

where $T_s^{r,q}$ is the q^{th} category of the r^{th} sensor corresponding to the s^{th} test of a given class *j*.

By definition, the Bayesian rule assigns $\tilde{T}$ to the class with the largest value of $P_i \times f_j(\tilde{T})$, where P_j $(j=1,...,m)$ represent the prior probabilities of each class. For the previous example introducing two classes, the following question is considered (Billard and Diday [2006]):

$$\text{Is } P_1 \times f_1(\tilde{T}) > P_2 \times f_2(\tilde{T}) \; ? \tag{10}$$

If the first term is greater than the second one, then the test $\tilde{T}$ is considered to belong to class 1; otherwise, the test is assigned to class 2. Although this example contains only two groups, this procedure can easily be extended to problems which have several classes. In fact, if there are *m* possible classes, one will have $m-1$ questions to assign a new test to a given class.

In addition, if prior probabilities are unknown, two choices are available to determine them:

- uniform prior probabilities based on the number of classes:

$$p_j=\frac{1}{m}, \qquad j=1,...,m \tag{11}$$

- prior probabilities based on the proportions observed in the training set:

$$p_j=\frac{n_j}{n} \quad \text{with} \quad n=\sum_{j=1}^{m}n_j \tag{12}$$

where *n* is the number of tests in the training set.

As already mentioned, tests are classified into different nodes according to "cut rules" and splitting criteria. To define each split, the most discriminant feature (sensor) must be used, that is, the feature capable to optimally separate tests into different classes. This "optimal" feature is called *cut variable* and it leads to the "purest" nodes of the Bayesian tree. The purity of a node is the number of tests that are correctly classified in it (by leave-one-out and/or bootstrap methods), where the classification is verified in relation to the class variable *C*. This number can be obtained by the Bayesian rule computed for each feature. The choice of the most discriminant *cut variable* will result in selecting the feature that minimizes the impurity measure i.e., the number of misclassified tests. Finally, the *cut variable* and its associated *cut value* will form the *cut rule*, which will be used to properly classify a new test (Song et al. [2006]).

2.3.2 Artificial Neural Networks

Neural networks have proven themselves as proficient classifiers and are particularly well suited for addressing non-linear problems. Given the non-linear nature of real-world phenomena, like the classification of dynamic tests, neural networks are a potential approach for dealing with this problem. Commonly, neural networks are adjusted or trained, so that a particular input leads to a specific target output (*supervised learning*). In this case, the network is adjusted based on a comparison of the output and the target, until the network output matches the target. Supervised learning is a machine learning technique for deducing mapping functions from a training dataset consisted of input-output pairs. The goal is to predict output values of the mapping function for any valid input after having seen some training examples (i.e., pairs of inputs and target outputs). To achieve this, the network must generalize from the training data to unseen situations in a "reasonable" way.

Mapping functions are obtained through an optimization scheme based on the evaluation of the mean-squared error (explained later). This scheme tries to minimize the average squared error between the network's output and the target value over all the training dataset pairs. In this chapter, a feed-forward multilayer perceptron neural network is used for classifying dynamic tests. Multilayer networks use a variety of learning techniques, the most popular being back-propagation. In this case, the output values are compared with the correct answer to compute the value of some predefined error-function and the error is then fed back through the network (Bezdek [1981]). Using this information, the algorithm adjusts the weights of each connection to reduce the value of the error function by some small amount. In this study, training automatically stops when generalization stops improving, which is indicated by an error increase in the validation samples. At this point, it is said that the network is "trained".

Let us consider, a one-hidden-layer MLP with *N* hidden neurons where the inputs are the dynamic tests T_i $(i = 1,...,n)$ which are symbolic representations of signals as

described in section 2. They are defined by $p \times k$ matrices, where p is the number of sensors and k is the number of categories. Like the BDT method, outputs are set to represent the target vectors (labels) corresponding to each class i.e., $\{1,2,...,m\}$. These outputs are transformed into a binary notation according to the number of classes used. Target vectors have m elements, where for each target vector, one element is 1 and the others are 0. This defines a problem where inputs are to be classified into m different classes.

The general formulation for a neural network can be written as follows: each input x_i (which can either be an input of the network or of a layer) is multiplied by adjustable weights denoted w_{il} before being fed to the l^{th} neuron in the hidden/output layers, yielding (Milligan and Cooper [1985]):

$$o_j = f\left(\sum_{i=1}^{N} w_{il} x_i + b_i\right) \quad j = 1,...,n \tag{13}$$

where o_j, b_i and f represents the output (either from a layer or from the network), the bias of each perceptron and the activation function, respectively.

To adjust weights properly, a general method for non-linear optimization called *gradient descent* is applied (Milligan and Cooper [1985]). Briefly, the derivative of the error function with respect to the network weights is calculated and the weights are then changed such that the error decreases (thus going downhill on the surface of the error function). Eq. 14 shows the expression to evaluate the updated weights of this network:

$$w_{il}(t+1) = w_{il}(t) - \eta \frac{\partial J}{\partial w_{il}} \tag{14}$$

where η is the learning rate, t is the iteration step and J is mean error for a perceptron, written as:

$$J = \frac{1}{2n} \sum_{j=1}^{n} (C_j - y_j)^2 \tag{15}$$

where C_j represent the desired outputs (targets) and y_j the observed outputs (evaluated by the neural network). For the simulations presented in this chapter, a free Matlab toolbox named Netlab developed by Bezdek [1981] was used. This toolbox allows performing several types of supervised multi-class classifications. The architecture of the NN consisted of a 20-neuron, one hidden-layer network using sigmoid activation function (hidden-layer) and linear function (output layer).

2.3.3 Support Vector Machines

Support Vector Machines (SVM) are a useful technique for data classification problems. As usual, the objective is to separate two different classes by a function which is induced from available examples (training dataset). SVMs were first suggested in the 1960s for

classification and have recently become an area of intense research due to developments in the techniques and theory coupled with extensions to regression and density estimation (Alves et al. [2015b]; van Overschee and de Moor [1996]). This technique came up from statistical learning theory and is based on the structural risk minimization principle. Commonly, SVMs are used for two-class classification problems. However, this can be extended from 2-class classifications to *m*-class classification problems by constructing m two-class classifiers. Thus, for multi-class SVM methods, either several binary classifiers must be constructed, or a larger optimization problem is needed. In the work of Hsu et al. [2002], a decomposition implementation for "all-together" methods, "one-against-all", "one-against-one" and Directed Acyclic Graph Support Vector Machines (DAG-SVM) were tested and compared.

In this section, a very brief review of SVM classification is given; further information can be found in references (Alves et al. [2015b]; van Overschee and de Moor [1996]). Let us consider $\{x_i, y_i\}_{i=1}^{n}$ a training dataset, where x_i are the input samples (symbolic representations of signals) and $y \in \{+1, -1\}$ the labels of classes and n the number of samples, respectively. According to Vapnik's original formulation, the hyperplane (HP) is defined as $wx + b = 0$, where x is a point lying on the HP, w determines the orientation of the HP, and b is the bias of the distance of the HP from the origin. If this HP maximizes the margin, then the following inequality is valid for all input data:

$$(b + w^T x_i) y_i \geq 1, \text{ for all } x_i,\ i = 1,2, \dots, n \tag{16}$$

The margin of the HP is equal to $2/\|w\|$, any training turples that fall on HP_1 or HP_2 (i.e., the sides defining the margin) are called support vectors. Thus, the problem is the maximization of the margin by minimizing $\|w\|/2$ subject to Eq.(16).

Lagrange multipliers $\alpha_i (\alpha_i > 0, 1 = 1, \dots, n)$ are used to solve $J_p = -\sum_{i=1}^{n} \alpha_i [(b + w^T x_i) y_i - 1] + \|w\|^2/2$. After minimizing J_p with respect to both $\boldsymbol{w}$ and b, the optimal values are given by: $w^* = \sum_{i=1}^{n} \alpha_i^* y_i x_i$. The so-called dual problem can be described as:

$$J_p(\alpha) = -\frac{1}{2} \sum_{i=1}^{n} \sum_{j=1}^{n} \alpha_i \alpha_j y_i y_j x_i x_j + \sum_{i=1}^{n} \alpha_i \tag{17}$$

Thus, the linear decision function is created by solving the dual optimization function, which can be obtained by:

$$f(x) = sgn\left(\sum_{i=1}^{n} \alpha_i^* y_i x_i x^T + b^* \right) \tag{18}$$

where a_i^* are the optimal Lagrange multipliers.

For input data with a high noise level, SVM uses soft margins that can be expressed as follows with the introduction of non-negative slack variables ξ_i, $i = 1, \dots, n$:

$$\left(b + \boldsymbol{w}^{*T} x_i\right) y_i \geq 1 - \xi_i, \text{ for } i = 1,2, \dots, n \tag{19}$$

To obtain the optimum separation HP, it should be minimized the $\psi = \mathrm{C}\sum_{i=1}^{n} \xi_i + \frac{1}{2}\|w\|^2$ subject to Eq. (19), where C is the penalty parameter. In the nonlinearly separable cases, the SVM maps the training points, nonlinearly, to a high dimensional feature space using kernel function $K(x_i, x) = \varphi(x_i).\varphi(x)$, where linear separation may be possible. The kernel function, $K(x, x_i)$, typically has multiple alternatives. In this chapter, the Radial Basis Function (RBF) is considered:

$$K(x_i, x) = \exp\left(-\|x_i - x\|^2 / 2g^2\right) \tag{20}$$

where $g \in R^+$ is a constant.

After a kernel function is selected, Eq. (20) becomes:

$$f(x) = sgn\left[\sum_{i=1}^{n} a_i^* y_i K(x_i, x) + b^*\right] \tag{21}$$

In general, RBF kernel is a reasonable first choice in training SVM, thus this kernel was used in this study. The selection of parameters C (the penalty term) and g (the basis width of the kernel) for the SVM model influences the classification accuracy significantly. In this work, an iterative algorithm was used during the validation phase to determine their optimal values. The SVM model in this chapter was implemented using the software LibSVM developed by Chang and Lin [2021].

3. Experimental applications and results

3.1 Simply supported beam

This section presents the experimental tests conducted at COPPE/UFRJ laboratory on a simply supported steel beam depicted in Figure 4.

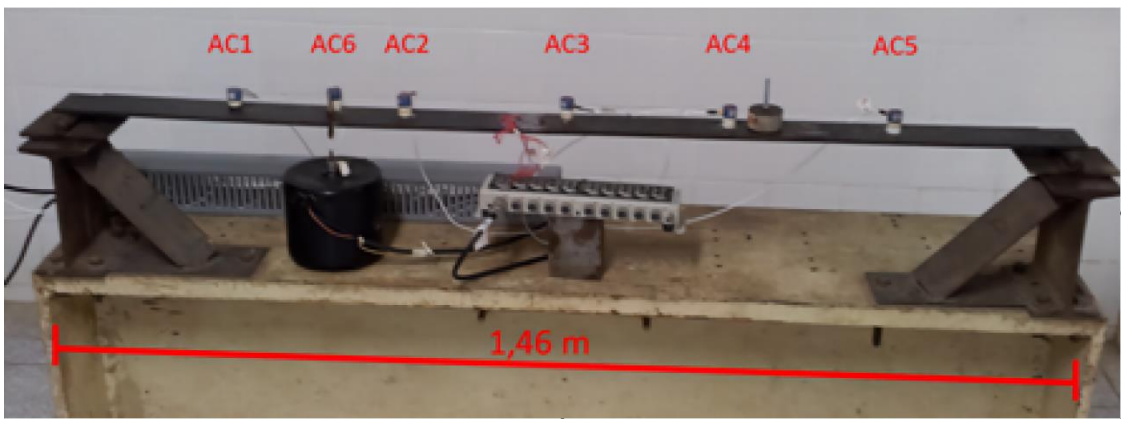

Figure 4: Instrumented steel beam.

The beam is 1.46 m long with rectangular cross-section (76,2 x 8,0 mm) and was instrumented with six piezoelectric accelerometers (PCB, 336C31). Data acquisition was carried out using Lynx ADS2002 equipment, which essentially is a conditioning/amplifier regulating system.

The present study considered two types of dynamic excitation: impact tests (using an impact hammer) and random vibration tests (using a shaker). For the impact tests, an impact hammer was used at each 10 seconds of the dynamic tests. The random excitation was applied throughout the duration of the tests. Six acquisition campaigns were performed, and, for each campaign, three tests were conducted. Thus, 18 dynamic tests recorded in total. Each one of them lasted for about 10 minutes with a sampling rate of 0,00025s (4000 Hz). This means that, for each test, 2.4 million values were recorded per sensor. Figure 5 shows an example of the dynamic tests performed under impact excitation (left) and random vibration (right).

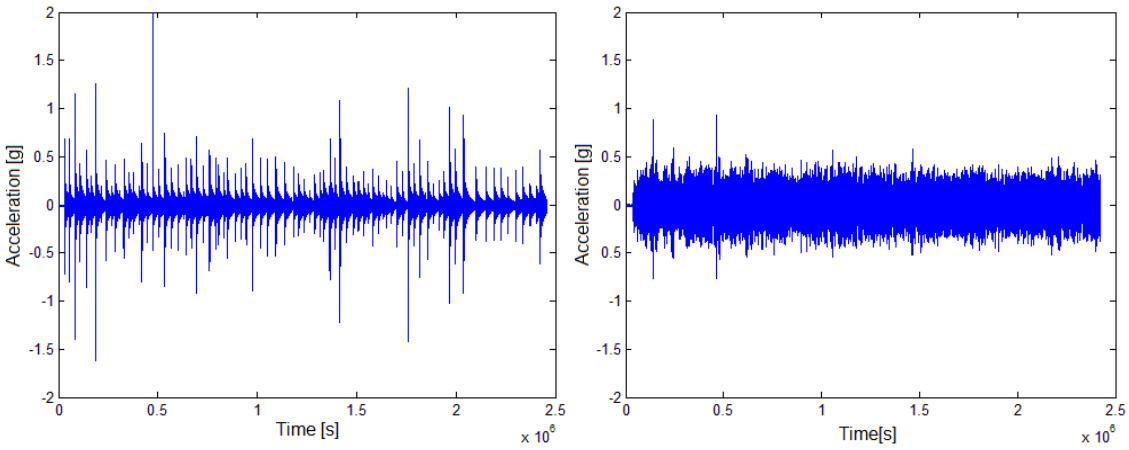

Figure 5: Example of vibration tests performed in laboratory: impact (left) and random (right).

The first campaign comprised the beam without damage. For the second campaign (Level 1), an eccentric mass (0.5 kg) was positioned at 102.7cm from the left support of the beam (between accelerometers AC5 and AC6). In fact, this second campaign was conducted so it could produce a reversible, non-permanent, “damage scenario”. The third campaign (Level 2) considered a small damage inflicted to the beam (a 12 mm round hole) imposed at the same position of the mass, which was removed beforehand. The fourth campaign (Level 3) consisted in increasing the hole to 16 mm. During the fifth and sixth campaigns (Levels 4 and 5), the hole was increased to 22.5 mm and 32 mm diameter, respectively. Figure 6 depicts the 12 mm hole imposed to the beam. It should be noted that these campaigns simulated very small structural damage, which makes this study even more challenging.

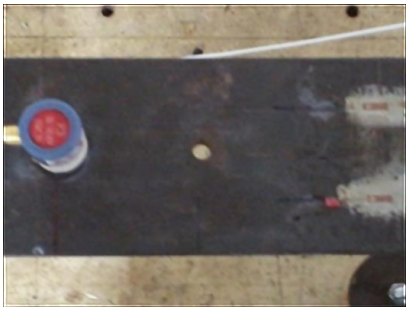

Figure 6: 12-mm hole imposed to the beam.

Modal identification was performed using Stochastic Realization Method algorithm (van Overschee and de Moor [1996]). Tables 1 and 2 present the mean values and respective standard deviations for the first five natural frequencies identified under impact and random vibration. Standard deviation values were rather low, which corroborates the overall quality of the modal identification procedure. In general, it is possible to observe a slight decrease of all five natural frequencies as damage increases. Exceptions occur and may be due to the modal identification procedure, mainly in the higher vibration modes where the signal/noise ratio is low. This agrees with results commonly presented in the literature. If one considers, however, confidence intervals i.e., mean values ± standard deviation, the values for natural frequencies from all damage scenarios would superpose themselves. Thus, from a statistical point-of-view, it is not possible to state that the deviation of these frequencies indicates a structural change. These results show that a more detailed analysis is necessary, but also considering a probabilistic approach, such as SDA.

Table 1. Frequency deviation according to each damage level (mean value and standard deviation - impact vibration).

Damage Level	Freq.#1	Freq. #2	Freq.#3	Freq.#4	Freq.#5
Undamaged	8,30 ± 0,170	33,31 ± 0,025	73,99 ± 0,011	134,90 ± 0,219	205,83 ± 0,135
Level 1	7,90 ± 0,196	31,40 ± 0,036	73,56 ± 0,062	131,63 ± 0,054	195,20 ± 0,023
Level 2	8,26 ± 0,060	33,24 ± 0,025	74,30 ± 0,013	136,23 ± 0,091	205,10 ± 0,041
Level 3	8,25 ± 0,064	33,21 ± 0,026	74,31 ± 0,0106	135,63 ± 0,158	205,57 ± 0,148
Level 4	8,25 ± 0,078	33,23 ± 0,056	74,66 ± 0,044	132,63 ± 0,112	205,90 ± 0,099
Level 5	8,23 ± 0,098	33,01 ± 0,064	74,19 ± 0,077	134,07 ± 0,165	204,30 ± 0,107

Table 2. Frequency deviation according to each damage level (mean value and standard deviation - random vibration).

Damage Level	Freq.#1	Freq. #2	Freq.#3	Freq.#4	Freq.#5
Undamaged	8,30 ± 0,170	33,31 ± 0,025	73,99 ± 0,011	134,90 ± 0,219	205,83 ± 0,135
Level 1	7,90 ± 0,196	31,40 ± 0,036	73,56 ± 0,062	131,63 ± 0,054	195,20 ± 0,023
Level 2	8,26 ± 0,060	33,24 ± 0,025	74,30 ± 0,013	136,23 ± 0,091	205,10 ± 0,041
Level 3	8,25 ± 0,064	33,21 ± 0,026	74,31 ± 0,0106	135,63 ± 0,158	205,57 ± 0,148
Level 4	8,25 ± 0,078	33,23 ± 0,056	74,66 ± 0,044	132,63 ± 0,112	205,90 ± 0,099
Level 5	8,23 ± 0,098	33,01 ± 0,064	74,19 ± 0,077	134,07 ± 0,165	204,30 ± 0,107

Mode shapes are omitted, since their study is not the focus of this chapter, but they followed a series of sinusoidal curves, as one should expect for simply supported beams.

3.1.1 – Results (unsupervised classification methods)

The procedure conducted henceforth in this chapter follows these steps:

1. Convert acceleration measurements into symbolic data (10-category histograms) as explained in section 2;
2. Use symbolic histograms (step 1) as inputs for the clustering techniques (hierarchy-agglomerative, dynamic clouds and FCM) considering different number of clusters (greater than 2);
3. Evaluate the optimal number of clusters using indexes *CH*, C^* and Γ applied to dynamic clouds and FCM methods (these are the only clustering methods in this chapter that require an initial number of clusters);
4. Retrieve partition of clusters obtained considering the optimal number of clusters (step 3).

Furthermore, it is important to emphasize that this entire procedure strongly depends on the quality of the input data. In this case, if accelerations measurements present any type of problem (bad sampling, missing data, incorrect measurement, etc.), the results obtained from the clustering methods will be compromised. Thus, it is imperative to assure, in firsthand, that the data used in the analysis is adequate.

The proposed approach is thus applied to accelerations measured during the experimental tests carried out in laboratory. Let us recall that the main objective here is to try to separate the six structural conditions (undamaged and damaged levels 1, 2, 3, 4

and 5) into six different clusters, using raw data (signals) only. Moreover, each type of forced vibration is considered separately.

The first simulations comprehend dynamic tests obtained under impact vibration only. As explained at the beginning of section 3, the proposed approach follows four steps. Once the clustering procedure is performed considering different number of clusters (in this case, it varied from 2 to 10), the indexes are evaluated.

Table 3 contains the first results. The last column shows the optimal number of clusters according to each index. It can be observed that both CH and Γ indicate 6 clusters (C^*, however, indicates 7). It can also be seen that all tests corresponding to structural conditions "undamaged", "damaged – level 1" and "damaged – level 3" are correctly classified. However, for the cluster "damaged – level 2", only one third of tests is classified properly. In general, results are very good and show that it is possible to extract information from raw data. Moreover, all clustering methods performed quite similarly. This does not mean, however, that all techniques will always yield the same results.

To attest the efficiency of the fuzzy c-means method, it is possible to evaluate the pertinence values, which quantify the certainty of the classification. If this index is 1, it means that the method is completely sure about its classification (which does not imply that the result is correct). Otherwise, if the pertinence value is close to 0, the method is "not sure" about its classification. In this sense, it is possible to observe that pertinence values for the classification showed in Table 3 are relatively high, validating the clustering procedure and, in this case, the results obtained.

Table 3: Percentages of correct classification (impact vibration).

(%)	Dynamic Clouds	Hierarchy Agglomerative	FCM	Pertinence value	Indexes
Undamaged	100	100	100	0,72	$CH = 6$
Level 1	100	100	100	0,92	$C^* = 7$
Level 2	33	33	33	0,79	$\Gamma = 6$
Level 3	100	100	100	0,83	
Level 4	100	0	100	0,81	
Level 5	66	66	66	0,74	
Average	**83**	**67**	**83**	-	

When it comes to random vibration tests, results are not as adequate as observed with impact vibration tests. Table 4 shows the percentages of correct classification for the six clusters and the respective pertinence values for the fuzzy c-means method. Once

more, both CH and Γ indicate 6 clusters but C^*, however, indicates 9. This latter tends to indicate a higher number of clusters, as observed in references (Billard and Diday [2006]; Cury and Crémona [2012]; Finotti et al. [2019]).

Table 4: Percentages of correct classification (random vibration).

(%)	Dynamic Clouds	Hierarchy Agglomerative	FCM	Pertinence value	Indexes
Undamaged	66	66	66	0,69	$CH = 6$
Level 1	33	33	66	0,63	$C^* = 9$
Level 2	33	100	33	0,58	$\Gamma = 6$
Level 3	66	33	66	0,67	
Level 4	100	100	100	0,77	
Level 5	33	33	33	0,61	
Average	**55**	**61**	**61**	-	

In this case, only one cluster is pure (for damage level 4). All other clusters are mixed, showing a difficulty for those methods to discriminate these structural states under random vibration. Although the classification ratios are lower, it is interesting to notice that the pertinence values also yield low scores (except for the cluster corresponding to level 4).

As a further analysis of this experimental application, the clustering procedures are carried out considering only damage levels 2, 4 and 5 (which corresponds to holes with 12, 22.5 and 32mm diameter). Once again, it should be noted that these campaigns simulated very small structural damage, which makes this study even more challenging.

Table 5 presents the classification ratios obtained considering impact vibration tests. It can be observed that the results improve significantly. Now, almost all classifications are perfect, except for the last cluster where only one third of the tests were classified incorrectly. In this analysis, all indexes indicated 4 clusters as the optimal partition.

Table 5: Percentages of correct classification (impact vibration).

(%)	Dynamic Clouds	Hierarchy Agglomerative	FCM	Pertinence value	Indexes
Undamaged	100	100	100	0,74	$CH = 4$
Level 2	100	100	100	0,91	$C^* = 4$
Level 4	100	100	100	0,84	$\Gamma = 4$
Level 5	66	66	66	0,74	
Average	**92**	**92**	**92**	-	

Similarly, even when random vibration tests are considered, classification results improved as well (see Table 6). In this case, however, index C^* has again pointed out a higher number of optimal clusters (5).

Table 6: Percentages of correct classification (random vibration).

(%)	Dynamic Clouds	Hierarchy Agglomerative	FCM	Pertinence value	Indexes
Undamaged	66	33	33	0,68	$CH = 4$
Level 2	100	100	66	0,76	$C^* = 5$
Level 4	100	100	100	0,77	$\Gamma = 4$
Level 5	66	66	66	0,74	
Average	**83**	**75**	**77**	-	

These last results can be partially explained by the fact that all these damage levels represent a more discriminative sequence of structural degradation. If damage levels are too similar, the proposed approach might not yield perfect classification scores. Nonetheless, it must be kept in mind that these results were obtained using raw data with no signal processing procedure whatsoever.

In general, better classification ratios were achieved when the structure was under impact vibration. In fact, when random vibration is considered, the transformation procedure to symbolic data (histograms) carries the noise from the excitation, thus misrepresenting the description of the dynamic test. Then, these "noisy" descriptions may mislead the clustering procedures into wrong classifications.

3.1.2 Results (supervised classification methods)

The procedure conducted henceforth follows two steps: i) transformation of acceleration measurements into symbolic data (10-category histograms) as explained in section 2; ii) use of these symbolic representations of acceleration measurements as inputs for the classification techniques (BDT, NN and SVM).

As previously mentioned, each dynamic test has 2.4 million points recorded per sensor. To make the classification analysis more robust, each test is subdivided into 24 'subtests' having 1.0 million points per sensor. Thus, instead of employing the 18 original dynamic tests for classification, now there are 18x24 = 432 tests to classify. This is important because all classification methods use training, validation and testing groups. If these groups are small or poorly rep-resented, the classification results may be affected.

The architecture of the NN consisted of a 20-neuron, one hidden-layer network using sigmoid activation function (hidden-layer) and linear function (output layer). This architecture was chosen after previous simulations, considering different numbers of layers and neurons. The learning rate was set to 0.001. For SVM classifier, the RBF kernel was used, and its parameters were chosen via an iterative algorithm during the validation phase.

The first simulations correspond to the impact excitation case and use only 30% of the set of dynamic tests for training, 10% for validation and 60% for testing. This means that the 432 tests are distributed over three groups: 130 tests for the training dataset, 43 tests for the validation dataset and 259 tests for the testing dataset. Since this

distribution can be randomly performed, 10000 simulations (10000 different groups of training, validation, and testing) are generated. This strategy was used following the work of references (Billard and Diday [2006]; Milligan and Cooper [1985]). Table 6 summarizes the results, showing the best, average, and worst classification ratios for each simulation. Considering the first column of this table, for example, it is possible to observe that the BDT has its best performance with 89% of correct classification. In average, it classifies 76% of tests correctly. Its worst performance occurs when it classifies only 51% of tests correctly among the 10000 simulations.

When the training dataset increases to 40%, the testing dataset reduces to 50%. Now, the probabilities of true detection improve significantly (average and worst ratios). Finally, when 50% of tests are used for training, all methods achieve their best results. It can be observed that the SVM reaches better results than the other two techniques. This might be because the SVM could better adjust "separation thresholds" for each damage state. Also, the BDT yield slightly worse probabilities of true detection compared to the NN. This could be explained by the fact that the BDT are not a true learning procedure. In fact, new tests are classified according to logical questions. If the six groups (corresponding to the damage levels) do not provide a meaningful representation of each structural state, the classification may be compromised.

In summary, what is important to extract from these simulations is the average values of true classification. Thus, both NN and SVM yield very satisfactory results. This shows that the pro-posed approach can classify structural conditions using only raw data recorded *in situ*.

Similarly, the simulations are performed using tests recorded under random excitation. Once again, as the training dataset increases, correct classification rates also improve. Moreover, both NN and SVM methods yield higher classification ratios.

In general, better classification ratios are obtained when the structure was under impact vibration. In fact, when random vibration is considered, the transformation procedure to symbolic da-ta (histograms) carries the noise from the excitation, thus misrepresenting the description of the dynamic test. Then, these "noisy" descriptions may mislead the classification procedures into wrong classifications. The authors acknowledge this drawback and further research are pointing into this direction.

Tables 7 and 8 omit, however, an important information: the number of occurrences for "worst" and "best" classification ratios. In all simulations, considering the impact excitation, the number of occurrences for worst ratios is lower than 10% of the total (967 simulations). This means that from all the 10000 simulations, less than 1000 reach their worst result. For the best ratios, 1654 simulations reach their best ratio (16,5%). The remaining simulations yield results between worst and best ratios with the corresponding average value presented in Table 6. Regarding the random excitation tests, results are slightly worse. In that case, 12,7% (1275) of the simulations yield the worst

ratio as only 9,8% (982) produce the best ratios. Again, the remaining results oscillate between worst and best ratios with the corresponding average value presented in Table 7.

Table 7. Probabilities of true classification using raw dynamic measurements (impact excitation).

	30% Tr., 10% V, 60% T			40% Tr., 10%V, 50% T			50% Tr., 10%V, 40% T		
Method	BDT	NN	SVM	BDT	NN	SVM	BDT	NN	SVM
Best (%)	89	100	100	94	100	100	94	100	100
Average (%)	76	85	88	68	90	91	67	93	95
Worst (%)	51	57	64	60	56	65	58	68	73

Table 8. Probabilities of true classification using raw dynamic measurements (random excitation).

	30% Tr., 10% V, 60% T			40% Tr., 10%V, 50% T			50% Tr., 10%V, 40% T		
Method	BDT	NN	SVM	BDT	NN	SVM	BDT	NN	SVM
Best (%)	88	96	96	90	99	99	91	100	100
Average (%)	65	83	84	65	89	89	64	91	94
Worst (%)	45	47	54	41	50	57	58	60	69

3.2 PI-57 motorway bridge

The PI-57 Bridge is a double-deck bridge located near the town of Senlis in France, crossing the Oise River, and carrying the A1 motorway, which connects Paris to Lille (Fig. 7). The bridge, a 116.50 m long, cast-in-place, post-tensioned segmental structure built in 1965, consists of three continuous spans of 18.00 m, 80.50 m, 18.00 m (Fig. 8).

Figure 7. PI-57 Bridge.

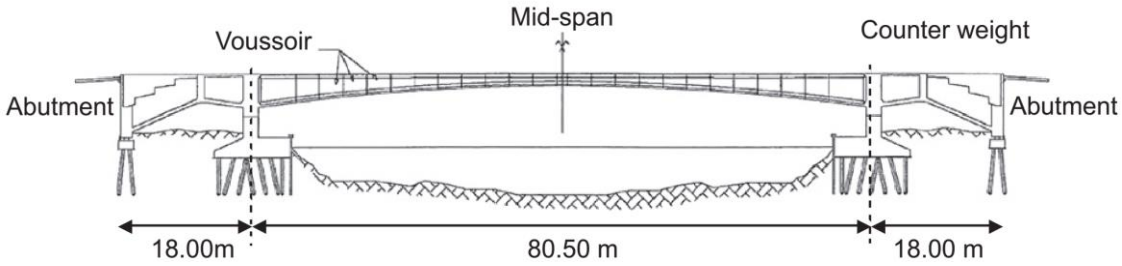

Figure 8. Bridge elevation.

The two lateral spans play the role of counterweights. This slender and elegant structure experienced various problems and distresses during and after its construction, resulting in localized cracking and increasing deflection in the central part. These problems were mainly due to insufficient prestressing because of lowering shrinkage and creep effects and a limited knowledge regarding thermal stresses at the time of construction. Because of the potential risk of cracking in the deck, numerical studies showed that the long-term integrity could be affected if corrective measures were not immediately taken. Based on these technical evaluations and considering the structure's importance, the concessionary motorway company (SANEF) decided to strengthen the two decks. Additional longitudinal prestressing (32,000 kN) would correct the lack of sufficient prestressing. The reinforcement works took place during the summer of 2009 (Alves et al. [2015b]). The strengthening consisted in reducing the tensile stresses under live loads to 1.5 MPa at the bottom part of the bridge cross-sections. These tensile stresses could reach 5.10 MPa at the mid-span cross-section. The external pre-stressing should induce at least 6.60 MPa compressive stresses (with a straight profile for the external prestressing cables). The anchorages were placed on the backside of the cross-girders located on the bridge piles. The total prestressing force has been evaluated to 32,000 kN corresponding to eight 19T15S cables (Fig. 9).

Figure 9. General view of the external prestressing cables installed in the bridge deck.

With this strengthening, the displacement at midspan decreased from 2.44 cm to 0.69 cm under SLS dead and live load effects. A small longitudinal displacement was expected (4.02 mm). No extra displacements occurred on the piles. To check on one hand the variability of the structural behavior due to thermal effects and on the other hand, to evaluate the efficiency of the strengthening procedure, a vibration-based monitoring was conducted: it consisted in installing accelerometers before and after reinforcement. The first campaign of measurements took place between October 15, 2008, and April 3, 2009. The second campaign, after reinforcement, started on October 15, 2009, and ended on April 3, 2010. Dynamic tests were performed under ambient excitation: the traffic was used as a source of excitation. Sixteen piezo-electric accelerometers (Bruel & Kjaer 4507B-005 with sensitivity 1 V/g, frequency range from 0.4 to 6000 Hz, maximum operational level 75g, temperature range from 54 to 100ºC) and seven temperature sensors (Pt100 class B) have been installed on the most deficient bridge deck (Lille/Paris – Figs. 10 and 11).

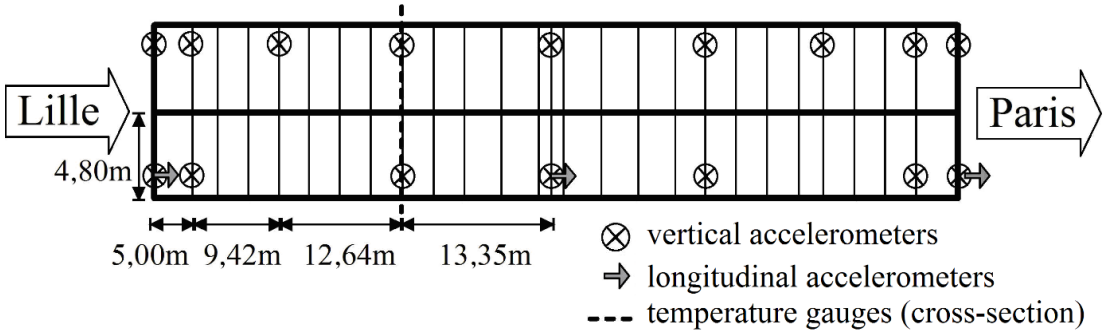

Figure 10. Plane view of the bridge with monitoring system.

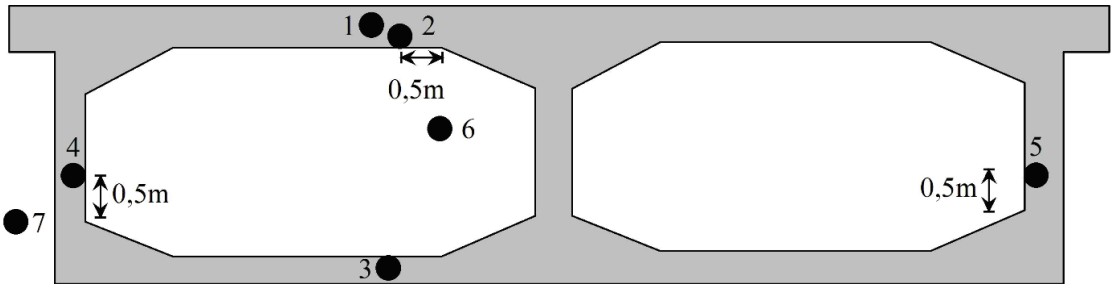

Figure 11. Location of the temperature sensors.

The data acquisition system was com-posed of two separate data acquisition systems. For the acceleration recording, a data programmable controller Gantner E-PAC DL was used and connected to an 8 GB USB flash drive. Data was transferred by a TCP/IP modem. For the temperature recording, a data logger Gantner IDL100 was used, and data was transferred by a GSM modem. Accelerations were filtered within the 0–30 Hz frequency range and sampling was set to 0.004 s during 5 min. To make the data processing amenable, structural data were only recorded every 3 h over a 24-h period and stored on a buffer hard disk. For the first campaign, 1174 tests have been recorded. The second campaign has had a total of 1316 tests recorded. Sixteen accelerometers were measuring vertical accelerations and three longitudinal accelerations (Fig. 10). The

instrumentation scheme allowed identifying flexural vertical modal shapes, torsional and longitudinal vibration modes. Temperature was measured on seven different locations across a bridge deck cross-section (Fig. 11). Table 9 regroups the first five natural frequencies, showing mean values and respective standard deviations before and after reinforcement. Standard deviation values were rather low, which corroborates the overall quality of the modal identification procedure. Once again, if one considers confidence intervals i.e., mean values ± standard deviation, the values for natural frequencies from the first and second campaigns would superpose themselves. Thus, from a statistical point-of-view, it is not possible to state that the deviation of these frequencies indicates a structural change. Again, these results show that a more detailed analysis is necessary, but also considering a probabilistic approach, such as SDA.

Table 9. Frequency deviation according to each damage level (mean value and standard deviation).

Structural condition	Freq.#1	Freq. #2	Freq.#3	Freq.#4	Freq.#5
Before reinforcement	2,23 ± 0,053	4,89 ± 0,173	6,84 ± 0,111	8,48 ± 0,173	11,00 ± 0,165
After reinforcement	2,29 ± 0,067	4,95 ± 0,088	6,93 ± 0,161	8,51 ± 0,135	11,08 ± 0,176

Fig. 12 depicts a typical set of temperature data for the gauges located in the mid-span section. These data were collected every hour, over a 24-h period, before the bridge strengthening (from November 2008 to April 2009). In this figure, the two lower curves are for temperature gauges located on the side of the bridge close to the other bridge deck (east) and outside the instrumented deck. As expected, this later temperature gauge has a larger variability than the other gauges. The highest curve is for the air temperature inside the girder box; T5 (lateral west) sensor is very close to the evolution of the inside air temperature with a time lag of 2–3 h. Sensors T1 and T2 are very close and the gauge depth in the concrete deck does not appear to be significant. There is also a time lag of approximately 2–3 h between the peak air temperature outside and the peaks for the inside and lateral west temperatures. Gauges T1, T2 and T3 are following very closely the time history of the inside temperature. In this study, the temperatures are used as a basis for comparison with the vibration information.

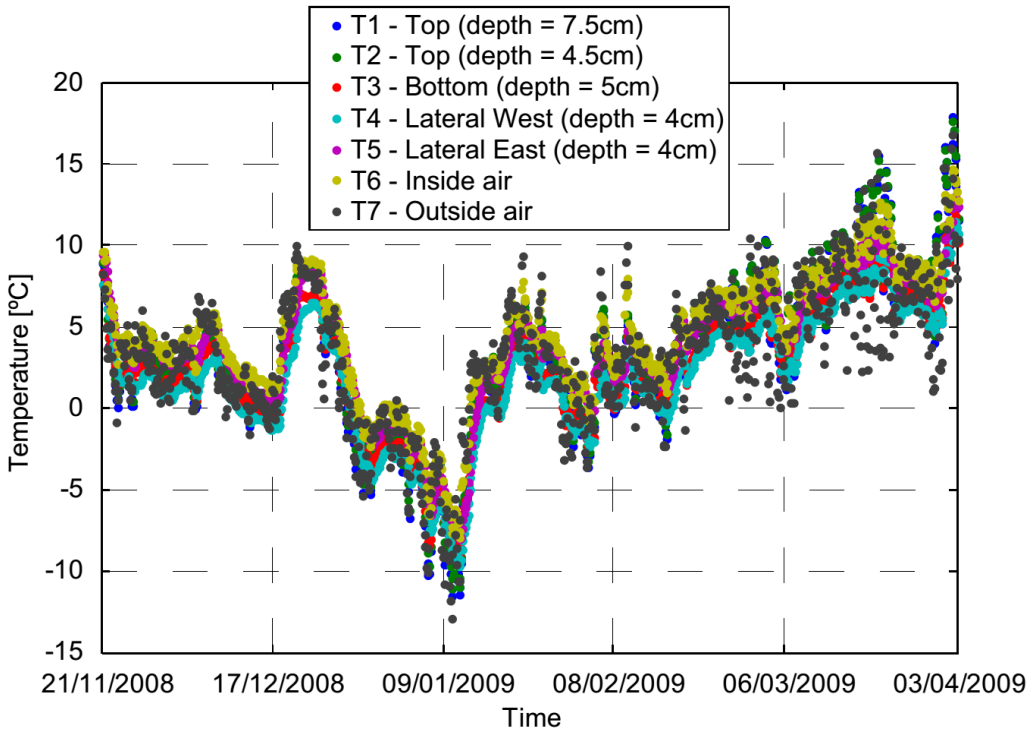

Figure 12. Time history of temperatures (before strengthening).

3.2.1 Results (unsupervised classification methods)

The procedure conducted henceforth in this section follows the steps described in section 3.1.1. Initially, the proposed approach is applied to all tests transformed into symbolic data (10-category histograms). The first campaign recorded 1284 tests as the second one collected 1476 tests. The objective of this study is to classify each test according to its campaign (before and after prestressing works). Thus, the target value must be intuitively ‘‘2’’ since two states (before and after strengthening) are expected to occur.

Table 10 shows the results obtained. In general, the percentages of correct classification are rather low for both dynamic clouds and hierarchy-agglomerative methods. Fuzzy c-means yield better results with moderate pertinence values. It must be noted that the proposed approach is being applied to the entire raw dataset of measurements, mixing dynamic tests registered upon different temperatures. This is directly reflected by the results of the indexes: in this analysis, all indexes indicate more than two clusters as the optimal partition. In fact, it turns out that some clusters might comprehend groups with similar profiles of temperature or traffic, for instance.

Table 10: Percentages of correct classification.

(%)	Dynamic Clouds	Hierarchy Agglomerative	FCM	Pertinence value	Indexes
Before	54	52	63	0,66	$CH = 4$
After	61	45	67	0,67	$C^* = 5$
					$\Gamma = 3$

Thus, a more detailed analysis can be conducted. Instead of using the entire dataset of tests from both campaigns, only data recorded during the same month are used. The objective now is to compare tests recorded in a month in 2008 with those registered at the same month in 2009 (for the months of January, February, March and April, the results correspond to the years of 2009 and 2010). The idea is to reduce, in some way, the uncertainties related to changes in temperature and due to traffic indirectly. For the month of October 2008 and 2009, 182 tests were recorder; for November 2008/2009, 444 tests; 458 in December 2008/2009; 455 in January 2009/2010; 386 in February 2009/2010; 439 in March 2009/2010 and 126 in April 2009/2010. Table 11 gathers the results obtained.

Table 11: Percentages of correct classification.

	Dynamic Clouds		Hierarchy-agglomerative		FCM			Indexes		
	Before	After	Before	After	Before	After	Pertinence Value	*CH*	C^*	Γ
Oct08/Oct09	100	100	100	100	100	100	0,88	2	2	2
Nov08/Nov09	56	58	54	75	67	67	0,65	3	4	3
Dec08/Dec09	65	69	59	52	62	70	0,71	3	3	2
Jan09/Jan10	63	54	53	62	67	67	0,64	2	3	3
Feb09/Feb10	59	63	63	55	65	69	0,60	2	4	2
Mar09/Mar10	61	57	74	61	70	69	0,70	2	4	3
Apr09/Apr10	57	58	67	58	67	68	0,60	2	3	2
Average	**66**	**66**	**67**	**66**	**71**	**73**	**0,68**	**-**	**-**	**-**

Overall, the classification results for the monthly-basis analysis are adequate and significantly better than those obtained using all tests concomitantly (Table 8). It is important to notice that for the months of October all tests were classified correctly. This result shows that it is indeed possible to extract useful information from raw vibration data. For the further months, the percentages are not as high. The fuzzy c-means method achieves better results with higher pertinence values compared to those of Table 8. It can also be observed that the average percentages of correct classification are also higher than those obtained in the previous simulation. Finally, indexes indicate optimal partitions equal or closer to two clusters. This corroborates the idea that the temperature variation plays an important role when it comes to the classification of structural behaviors.

3.2.2 Results (supervised classification methods)

The procedure carried out in this section mirrors the steps described in section 3.1.2.

The results show that both NN and SVM are again more robust for classifying tests compared to BDT. In general, NN achieves higher rates of correct classification (greater than 85%). In this study, it is noted that increasing the number of tests in the training group did not significantly alter the rates of correct classification. This may

indicate that the proposed approach is sufficiently discriminative, even if few tests are used in the learning phase.

Like the previous study, Table 12 does not show the number of occurrences for the most extreme classification ratios (worst and best). In this case, 14,8% (1483) of the simulations yield the worst ratio and 7,3% (734) produce the best ratios. Again, the remaining results oscillate be-tween worst and best ratios with the corresponding average value presented in Table 12.

Table 12. Probabilities of true classification using signals for all dynamic tests.

	30% Tr., 10% V, 60% T			40% Tr., 10%V, 50% T			50% Tr., 10%V, 40% T		
Method	BDT	NN	SVM	BDT	NN	SVM	BDT	NN	SVM
Best (%)	74	87	86	80	89	90	85	92	92
Average (%)	65	75	74	69	75	76	74	78	77
Worst (%)	31	46	749	45	58	46	52	60	55

However, a more detailed analysis can be conducted. Instead of using the entire dataset of tests from both campaigns, only data recorded during the same month are used. The objective now is to compare tests recorded in a month in 2008 with those registered at the same month in 2009 (for the months of January, February, March and April, the results correspond to the years of 2009 and 2010). The idea is to reduce, in some way, the uncertainties related to changes in temperature and due to traffic. For the month of November 2008 and 2009, 444 tests were record-ed; 458 in December 2008/2009; 455 in January 2009/2010; 386 in February 2009/2010; 439 in March 2009/2010 and 126 in April 2009/2010.

Figure 13(a) summarizes the results obtained when only 30% of the tests (each month, separately) are retained in the training group. Once more, both NN and SVM are the most efficient classification methods. Indeed, the coupling of these methods to the symbolic representation of signals (histograms) is more sensitive to external effects (traffic, temperature, wind, etc.). However, using a monthly basis analysis, one observes the reduction of these effects, yielding better classification ratios. Figures 13(b) and 13(c) show the mean values obtained for each classification method considering the training groups with 40% and 50% of the tests, respectively. In general, the average classification rates are in the range of 80-90% considering the three methods and the three configurations of testing/training groups.

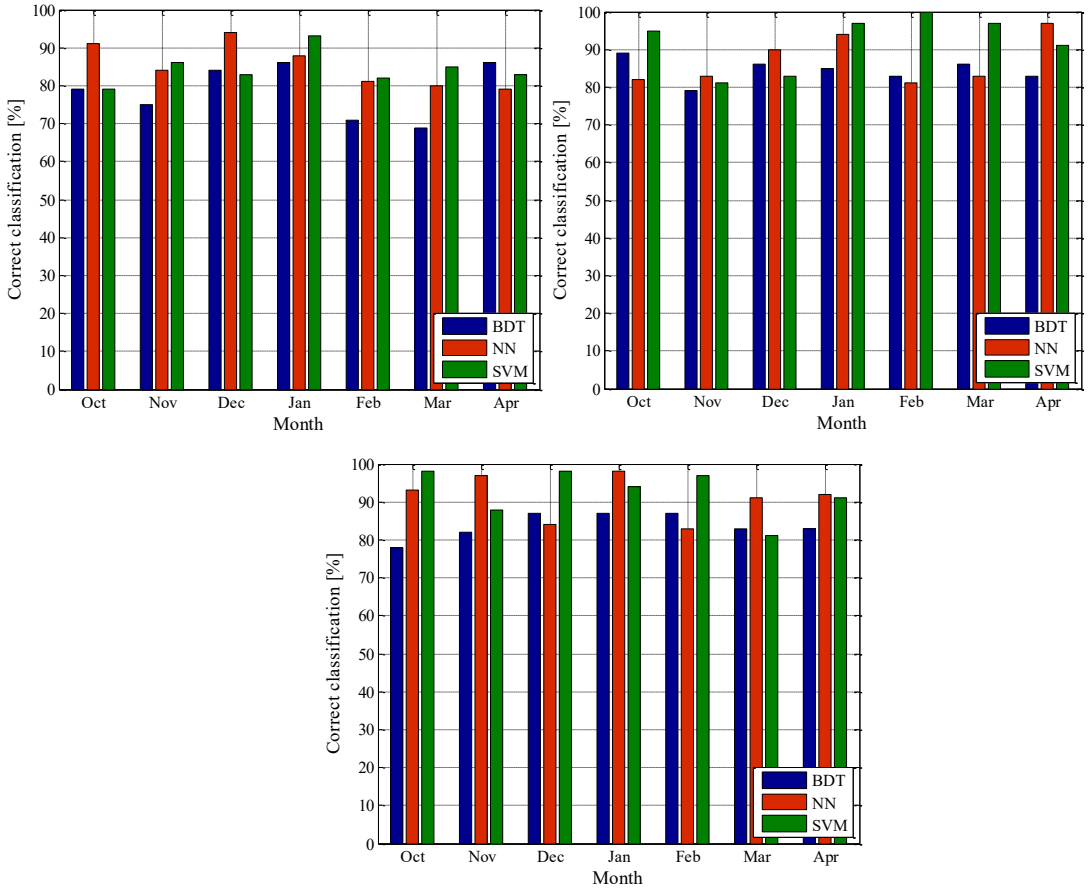

Figure 13. Average rates of correct classification. a) 30% training, 10% validation, 60% testing; b) 40% training, 10% validation, 50% testing; c) 50% training, 10% validation, 40% testing.

4. Final remarks and recommendations

This chapter presented a collection of approaches based on the coupling of Symbolic Data Analysis with three unsupervised and supervised classification methods. The main goal was to discriminate different structural behaviors using only raw information for feature extraction.

To attest the robustness of the proposed approaches, an experimental application performed at COPPE Structural Laboratory was studied. This application comprised a simply supported beam instrumented with six accelerometers, tested under two different excitation conditions: impact and random vibration. Moreover, six damage scenarios were considered: one undamaged scenario and five others corresponding to holes with different diameters. The main objective was to discriminate those structural states using only measured accelerations. Results obtained showed that the SDA methods were quite efficient to classify and discriminate structural modifications, even when raw data are used. In addition, better results were obtained when impact excitation was considered.

Furthermore, this chapter presented an experimental application concerning reinforcement works performed at a motorway bridge in France. Dynamic tests were

registered under two structural states – before and after reinforcement. The results obtained showed that the SDA methods were efficient to classify and to discriminate structural modifications considering raw vibration data. To consider the effects of temperature variation, a more detailed study was conducted: instead of using the entire dataset of tests from both campaigns, only data recorded during the same month were used. In that case, results have improved significantly, showing evidence that this environmental effect plays an important role in the field of SHM.

In summary, the proposed combined approaches can be applied to continuous structural monitoring analyses when:

- reference behaviors are unknown → unsupervised learning methods. This is major advantage when SHM are performed over the years and a constant attention is necessary when it comes to structural safety. The main drawback might be the number of false positive alarms due to the lack of proper "learning" of the structure's behavior.
- one (or more) reference behaviors are known → supervised learning techniques. The main advantage is to provide the machine learning techniques a comprehensive database about the structure's dynamic behavior. Then, it allows mitigating the number of false positive alarms. The main drawback, on the other hand, is that it is not always possible to know the structure's actual (or former) condition *a priori*.

Acknowledgments

The authors would like to thank CAPES (Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Finance Code 001), CNPq (Conselho Nacional de Desenvolvimento Científico e Tecnológico – Grant 304329/2019-3), FAPEMIG (Fundação de Amparo à Pesquisa do Estado de Minas Gerais – Grant PPM-00001-18), and CIDENG-CNPq (Research Group on Data Science in Engineering). Moreover, the authors would like to thank to IFSTTAR (former LCPC - Laboratoire Central des Ponts et Chaussées) and SNCF (Société Nationale des Chemins de fer Français) - project 01V0527 RGCU "Evaluation dynamique des ponts" - for the data used in this chapter.

References

A. Alvandi and C. Cremona. Assessment of vibration-based damage identification techniques. Journal of Sound and Vibration, volume 292, no. 1–2, pages 179–202, 2006. DOI: 10.1016/j.jsv.2005.07.036.

V. Alves, A. Cury, N. Roitman, C. Magluta, and C. Cremona. Novelty detection for SHM using raw acceleration measurements. Structural Control and Health Monitoring, volume 22, no. 9, 2015a. DOI: 10.1002/stc.1741.

V. Alves, A. Cury, and C. Cremona. On the use of symbolic vibration data for robust structural health monitoring. Proceedings of the Institution of Civil Engineers -

Structures and Buildings, volume 169, issue 9, pages 715–723, 2015b. DOI: 10.1680/jstbu.15.00011.

J. C. Bezdek, Pattern Recognition with Fuzzy Objective Function Algorithms. Boston, MA: Springer US, 1981. DOI: 10.1007/978-1-4757-0450-1.

L. Billard and E. Diday, Symbolic Data Analysis. Chichester, UK: John Wiley & Sons, Ltd, 2006. DOI: 10.1002/9780470090183.

R. de A. Cardoso, A. Cury, F. Barbosa, and C. Gentile. Unsupervised real-time SHM technique based on novelty indexes. Structural Control and Health Monitoring, volume 26, no. 7, 2019. DOI: 10.1002/stc.2364.

A. A. Cury, C. C. H. Borges, and F. S. Barbosa. A two-step technique for damage assessment using numerical and experimental vibration data. Structural Health Monitoring, volume 10, no. 4, pages 417–428, 2011. DOI: 10.1177/1475921710379513.

A. Cury and C. Crémona. Assignment of structural behaviours in long-term monitoring: Application to a strengthened railway bridge. Structural Health Monitoring, volume 11, no. 4, pages 422–441, 2012. DOI: 10.1177/1475921711434858.

S. W. Doebling, C. R. Farrar, M. B. Prime, and D. W. Shevitz. Damage identification and health monitoring of structural and mechanical systems from changes in their vibration characteristics: A literature review. Los Alamos, NM, 1996. DOI: 10.2172/249299.

R. P. Finotti, A. A. Cury, and F. de S. Barbosa. An SHM approach using machine learning and statistical indicators extracted from raw dynamic measurements. Latin American Journal of Solids and Structures, volume 16, no. 2, 2019. DOI: 10.1590/1679-78254942.

S. J. S. Hakim and H. A. Razak. Modal parameters based structural damage detection using artificial neural networks - a review. Smart Structures and Systems, volume 14, no. 2, 2014. DOI: 10.12989/sss.2014.14.2.159.

U. Kroszynski and J. Zhou. Fuzzy Clustering - Principles, Methods and Examples. 1998.

T. S. Madhulatha. An overview on clustering methods. volume 2, no. 4, pages 719–725, 2012.

N. Martins, E. Caetano, S. Diord, F. Magalhães, and Á. Cunha. Dynamic monitoring of a stadium suspension roof: Wind and temperature influence on modal parameters and structural response. Engineering Structures, volume 59, pages 80–94, 2014. DOI: 10.1016/j.engstruct.2013.10.021.

G. W. Milligan and M. C. Cooper. An examination of procedures for determining the number of clusters in a data set. 1985.

J. J. Moughty and J. R. Casas. A state-of-the-art review of modal-based damage detection in bridges: Development, challenges, and solutions. Applied Sciences (Switzerland), volume 7, no. 5, 2017. DOI: 10.3390/app7050510.

P. van Overschee and B. de Moor. Subspace Identification for Linear Systems. Boston, MA: Springer US, 1996. DOI: 10.1007/978-1-4613-0465-4.

C.W. Hsu, C.-J. Lin. A comparison of methods for multi-class support vector machines, *IEEE Transactions on Neural Networks*, 13, 415-425, 2002.

C.C. Chang, C.J. Lin, LIBSVM: a library for support vector machines, 2021. https://github.com/cjlin1/libsvm.

J. Santos, C. Crémona, and P. Silveira. Automatic Operational Modal Analysis of Complex Civil Infrastructures. Structural Engineering International, volume 30, no. 3, 2020. DOI: 10.1080/10168664.2020.1749012.

Y.-S. Song, D.-C. Park, C. N. Tran, H.-S. Choi, and M. Suk. Fuzzy C-Means Algorithm with Divergence-Based Kernel. 2006. DOI: 10.1007/11881599_12.

S. Wang. Iterative modal strain energy method for damage severity estimation using frequency measurements. Structural Control and Health Monitoring, volume 20, no. 2, pages 230–240, 2013. DOI: 10.1002/stc.495.

Y. Xia, B. Chen, X. Q. Zhou, and Y. L. Xu. Field monitoring and numerical analysis of Tsing Ma suspension bridge temperature behavior. Structural Control and Health Monitoring, volume 20, no. 4, pages 560–575, 2013. DOI: 10.1002/stc.515.

Y. Zhou, M. Wahab, N. Maia, L. Liu, and E. Figueiredo, Data Mining in Structural Dynamic Analysis. Singapore: Springer Singapore, 2019. DOI: 10.1007/978-981-15-0501-0.