Data-driven machine learning for pattern recognition and detection of loosening torque in bolted joints

Jefferson S Coelho, Marcela R Machado, Maciej Dutkiewicz, Rafaël Teloli

▶ **To cite this version:**

Jefferson S Coelho, Marcela R Machado, Maciej Dutkiewicz, Rafaël Teloli. Data-driven machine learning for pattern recognition and detection of loosening torque in bolted joints. Journal of the Brazilian Society of Mechanical Sciences and Engineering, 2024, 46 (75), pp.46 - 75. 10.1007/s40430-023-04628-6 . hal-04584328

HAL Id: hal-04584328
https://hal.science/hal-04584328v1

Submitted on 23 May 2024

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Data-driven machine learning for pattern recognition and detection of loosening torque in bolted joints

Jefferson S. Coelho[1*], Marcela R. Machado[1,2†], Maciej Dutkiewicz[2†] and Rafael de O Teloli[3†]

[1*]Department of Mechanical Engineering, University of Brasília, Campus Universitário Darcy Ribeiro, Brasília, 70910-900, Distrito Federal, Brazil.

[2]Faculty of Civil, Environmental Engineering and Architecture, Bydgoszcz University of Science and Technology, 85-796, Bydgoszcz, Poland.

[3]Département Mécanique Appliquée, Université de Bourgogne Franche-Comté, , Besancon, Bourgogne Franche-Comté, France.

*Corresponding author(s). E-mail(s): jeffersoncoelho@ufam.edu.br;

Contributing authors: marcelam@unb.br; macdut@pbs.edu.pl; rafael.teloli@femto-st.fr ;

†These authors contributed equally to this work.

Abstract

One of the primary challenges associated with bolts, frequently used in assembly structures and systems, is the issue of torque loosening. This problem can arise due to shock and vibration, potentially leading to significant damage and structural failure. The difficulty in identifying and monitoring torque loosening arises from the variability and nonlinear effects present in bolted joints. In this paper, we proposed

a machine-learning algorithm architecture designed for pattern recognition, detection, and quantification of torque loosening in bolted joints. This approach combines unsupervised and supervised machine learning algorithms to address the challenges of assessing the bolt torque loosening issue. Our algorithm utilises a damage index, calculated from the frequency response of the jointed system using the Frequency Response Assurance Criterion, as input data for the unsupervised-supervised classification algorithm. This classification ML algorithm effectively identifies and categorises instances of torque loss by analysing indirect vibration measurements, even in situations where the bolted system's state is unknown. Additionally, we introduce a regression algorithm to quantify torque loosening levels. The results obtained from our proposed machine learning algorithm to overcome torque loosening in bolted joints show that the inherent uncertainties of a data-driven approach intrinsically influence torque-related issues. This assessment is based on experimental raw data collected under diverse test conditions for the bolted structure. We employ a range of validation and cross-validation metrics to evaluate the effectiveness and accuracy of these ML algorithms in detecting and diagnosing torque-related issues. These metrics play a crucial role in assessing the algorithms' efficiency and precision in determining the state of the bolted connection and the corresponding torque levels with associated uncertainty quantification.

Keywords: Bolt tightening, Damage diagnosis, Uncertainty quantification, Torque loosening detection, Unsupervised-supervised machine learning, Regression

1 Introduction

Structural are usually connected using bolts or fasteners. Bolts are a popular choice because they offer several benefits, such as preventing movement ensuring stability, and the ability to disassemble, reuse, and maintain structural integrity. However, bolt systems require a preload or torque level to prevent undesired movement. Unfortunately, this torque can loosen due to various causes, such as shock, vibration, inadequate tightening, and fracture. Reducing torque can lead to changes in the dynamic system feature, resulting in serious damage and structural failure, some of which can be catastrophic [1, 2]. As a result, constant and systematic maintenance is necessary throughout the system's life cycle. This costly and often dangerous operation requires regular checks of each bolted joint. Therefore, identifying and detecting loosening before failure remains a significant challenge in various engineering fields.

Over the years, significant advancements have been made in detecting loosening in bolted structures [3, 4]. These techniques are in situ inspection,

computational vision, and sensor-based techniques. In-situ inspection techniques, such as using a torque wrench and hammer, can be visual or mechanical [5, 22, 42]. Digital cameras or images are used in computer vision-based techniques, while the vibration-based method, wave propagation [43–45], acoustoelastic effect-based method, piezoelectric sensor-based methods, and impedance-based method are used in sensor-based techniques [5]. Machine Learning (ML) algorithms have recently been utilised to detect and monitor bolt torque loosening. These algorithms are categorised into supervised and unsupervised classification learning techniques and regression algorithms. The supervised learning technique creates a model from a set of labelled training data using previously known input and output values, and it is subdivided into classification and regression problems. Sousa et al. [46] accurately assess the damage of a beam reinforced by masses from its spectral response using multi-class supervised machine learning algorithms. The authors used ML to classify the beam's damage, where the methodology involves experimental measuring and numerical calculation of the dynamic features, such as natural frequency and frequency response function, to construct two DIs. In contrast, the unsupervised learning approach does not require target class labels in the training data[6]. The regression algorithms help in defining the relationship between labels and data points.

In situ inspection techniques to monitor torque loosening have been explored by Zhou et al. [7], where the authors describe a study that used percussive methods and machine learning to detect loosening in bolts. The experiment was conducted on a four-bolt steel beam-column joint, where laser Doppler vibrometry was used to capture the vibration information of the test bolt, while microphones collected acoustic sounds generated by an automatic hammer. The authors then transformed the reconstructed sound database into spectrograms and trained a 2D-CNN to identify bolts' loosening conditions. Wang and Song [8] presented a novel one-dimensional training interference capsule neural network (1D-TICapsNet) to process and classify percussion-induced sound signals to detect bolt early looseness in two steel pieces tightened using four bolts. They also employed in [9]the multifractal analysis and joint mutual information maximization method to extract feature sets and detect bolt loosening using the gradient boosting decision tree (GBDT) algorithm. Tran et al. [12] investigated the application of a deep convolutional neural network (DCNN) algorithm to detect and estimate looseness in bolted joints using a laser ultrasound technique. Zang et al. [10] presented a method of detecting screw loosening in iron plates based on audio classification using SVM. Kong et al. [11] proposed a new approach to identify bolt clearance levels in a twelve-bolt subsea flange using an ML model with the decision tree method, similar to the percussive diagnostic techniques used in clinical examinations.

Computer vision-based machine learning techniques to torque loosening were proposed by Gong [13]. In [18], the authors use a combination of deep learning algorithms and geometric image theory for detecting loosened bolts in a steel

pedestal through vision-based bolt loosening. The method uses a faster regional convolutional neural network (Faster-RCNN) and a waterfall pyramid network (CPN) algorithm. Similar research has been conducted by Zhang [16] and Yu [14] using the Faster R-CNN and single shot multibox detector (SSD) algorithm, respectively, for detecting bolt loosening angles. Pham et al. [15] used synthetic images of bolts generated from a graphical model to train a deep learning model based on the Region-based Convolutional Neural Network (R-CNN) algorithm for detecting loose bolts. Ramana et al. [17] used machine learning techniques, including the Viola-Jones algorithm and SVM, to detect loosened bolts on a steel I-section. Similarly, Chan et al. [19] used the Hough transform and SVM to build a classifier for detecting loosened bolts.

Several investigations have employed various techniques and methods to detect anomalies and identify bolt loosening in engineering structures. For instance, Razi et al.[20] have utilized sensor-based techniques, machine learning (ML) based on wave propagation, and modal methods. In the study by Ziaja et al. [21], elastic wave propagation was employed to detect anomalies in the prestressed connections of engineering structures, utilising a combination of artificial neural networks (ANN). Eraliev et al.[22] detected and identified loosening bolts in a multi-bolt structure using seven ML algorithms, namely Random Forest, Bagged Trees, Decision Tree, Kneighbor, Linear Discriminant Analysis, SVM, and XGBoost. The author utilised the Short-Time Fourier Transform (STFT) method for feature extraction from acquired vibration data. Miguel et al.[23] observed the loss of tightening torque in bolted joints by employing modal parameters. Teloli et al. [35] utilised two probabilistic ML methods, namely the Gaussian mixture model (GMM) for damage detection and Gaussian process regression (GPR) for quantifying loosening torque in lap-joint structures. Chen et al. [24] proposed a diagnostic method for detecting looseness in fan foundation bolts based on the mixed domain characteristics of excitation response and multiple learning. The study employed the K-weight nearest neighbour classifier (WKNNC) to identify slacks. Zhuang et al. [25] employed the acoustoelastic effect-based method along with several ML algorithms such as the recurrent neural network LSTM, one-dimensional WideResnet40_2, one-dimensional Densenet121, XGBOOST tree classification model, LightGBM, and the SAX-VSM algorithm. Wang et al.[26, 27] proposed the Siamese Double-path CapsNet (SD-CapsNet) and the Genetic Algorithm-based Least Square Support Vector Machine (GA-based LSSVM) for bolt loosening detection, using piezoelectric sensor-based methods. Zhou et al.[28] applied an impedance-based method using the Graph convolutional networks (GCN) model in another approach. Hence, these studies have utilised various ML algorithms to tackle complex real-world problems in diverse applications. Most proposed techniques or processes combining structural health monitoring (SHM) with ML are based on hybridising multiple interacting numerical procedures [60]. Consequently, this complexity often poses challenges when implementing an effective solution.

This study builds upon existing work in bolted joint analysis, particularly in recognising and detecting loosening torque from a structure's dynamic response. Prior studies have primarily relied on machine learning algorithms to assess bolted joint integrity, utilising modal parameters such as natural frequency and damping. However, these studies have typically overlooked critical factors such as data variability, noise, and spectral signals, which are of utmost importance when dealing with bolted joints. It is worth noting that modal parameters, although valuable, exhibit limited sensitivity to variations compared to measurement uncertainties, particularly at higher torque levels. In our approach to recognising, detecting, and quantifying bolt-loosening torque, we have departed from the conventional path by directly leveraging the raw spectral signals obtained from experimental tests to estimate a damage index. Then, this DI is used as input for our machine-learning algorithms, which avoid the need for prior operational modal analysis and incorporate the inherent variability within the experimental measurements when estimating loosening torque. To further enhance the analysis, several ML techniques were employed. The ML classifiers have been designed to withstand variabilities in raw data, in addition to noise variations, considering the influence of assembling and disassembling the bolted structure during experimental tests, as documented in [35]. The present approach simplifies the analysis by eliminating the need to evaluate the most sensitive features for extraction, introducing the concept of FRAC in obtaining the damage index, which plays a crucial role in our algorithm's methodology. As a result, the main contribution of our ML algorithm architecture for pattern recognition, detection, and quantification of loosening torque in bolted joints lies in its utilisation of spectral raw signals from experimental tests. This approach not only accounts for intrinsic variabilities in torque estimation but also quantifies the associated uncertainty, enhancing the robustness and accuracy of our results.

Towards this background, this paper is structured as follows: section 2 provides a description of the algorithm developed in this work for bolt loosening detection. Section 3 introduces the data-driven processing carried out on the structure being analysed, the feature selection and pattern recognition framework. Further details are also given on how the data from healthy conditions and damage was divided up for the learning stages. Then, section 4 focuses on developing the machine learning techniques used in the work to detect the loss of tightening torque, specifying the hyperparameters used and the different performance metrics analysed, allowing a comparison between methods. After detecting whether or not there is a loss of tightening torque, a natural step is to quantify it - this is discussed in section 4. Finally, section 5 reports the final remarks on expected contributions and proposes the next steps for future work.

2 Algorithm Framework

The proposed framework to pattern, detect and quantify the loosening bolt torque is presented in the flowchart of Figure 1. It includes: data acquisition (step 1), an unsupervised stage comprising data processing (step 2), feature selection (step 3), pattern recognition and clustering (step 4) - all these steps constitute the Data-Driven Processing and Pattern Recognition section of the methodology, which is addressed in section 3. Then, these steps are followed by data splitting (step 5). In the supervised stage, classification ML algorithms (steps 6 and 7) are used to detect and evaluate torque loosening (see section 4), and regression (steps 6 and 7) is employed for torque quantification (see section 5). At the end (step 8), the algorithm provides information about the torque state based on the classification and regression algorithm outcomes.

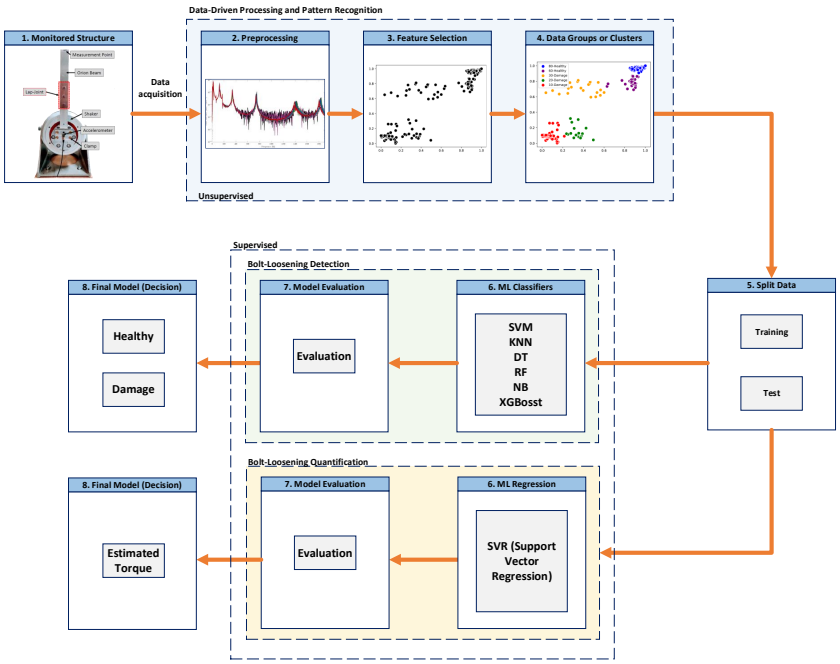


Fig. 1: Flowchart of a semi-supervised ML algorithm and the Pattern Recognition methodology to bolt loosening monitoring.

The structure proposed in this work can be established using the following algorithm:

Algorithm 1 Loosening bolt torque detection and quantification algorithm

1: **Data Acquisition:** Collect FRF ($H(\omega)$) data from monitored structure.
2: **Data Processing:** Converts raw experimental data into a normalised value using the FRAC Damage Index.
$$FRAC_{ij} = \frac{\left\| H_{ij}^{d}(\omega)\left(H_{ij}^{u}(\omega)\right)^{*} \right\|^{2}}{\left[H_{ij}^{u}(\omega)\left(H_{ij}^{u}(\omega)\right)^{*}\right]\left[H_{ij}^{d}(\omega)\left(H_{ij}^{d}(\omega)\right)^{*}\right]}$$
3: **Feature Selection:** Select a subset of features (variables, predictors) for use in model construction.
4: **Pattern Recognition and Clustering:** Use the K-means unsupervised algorithm to group the data into distinct clusters.
$$J = \sum_{i=1}^{n} min_{k} \left(\left\| x_i - \mu_k \right\|^2 \right)$$
5: **Data Splitting:** Split the dataset into training (70%) and testing (30%).
6: **Classification ML algorithms:** Applied ML algorithms for detection (SVM, K-NN, RF, NB, DT and XGBoost).
7: **Model Evaluation:** Calculate the Performance Measure for the ML classifiers (Cross-validation, Accuracy, Precision, Recall, and F1-Score).
8: **Regression**: Applied SVR-supervised regression for quantification.
$$Minimize: \quad \frac{1}{2}\left\|w\right\|^{2} + C\sum_{i=1}^{n}\left(\xi_i + \xi_i^{*}\right)$$
9: **Bolt-Loosening Quantification:** Quantifies the extent of bolt loosening using the SVR algorithm.
10: **Final decision:** Information about torque state based on classification and regression algorithm outcomes.

Each algorithm's steps are briefly detailed in the following:

Data-Driven Processing and Pattern Recognition:

1 **Data Acquisition:** The first step involves collecting data from the monitored structure, corresponding to obtaining the experimental raw data - they are available in [36]. The input data considered consists of 360 realisations from several experimental runs.

2 **Data Pre-processing:** The utilization of a data-driven involves a crucial step - understanding and examining its inherent features before integrating it into the machine learning model for accurate classification and estimation. Specifically, our case involves employing the spectrum response of the structure, which addresses limitations encountered when relying solely on modal parameters such as natural/resonant frequency and modal shape. These parameters exhibit sensitivity to uncertainties and lack the ability to detect certain types of damage effectively. To address these limitations, we incorporate spectrum analysis, necessitating additional pre-processing steps outlined in this step. It becomes imperative as some resonant frequencies in the spectrum display limited sensitivity to torque loosening, particularly evident in higher vibration modes. This step, also known as

feature extraction, involves transforming the original data variables to create a new dataset, as described by Bishop (2006) [39]. This data processing step converts raw experimental data into a normalised value using the DIs calculated with Eq. 1. The 360 DIs are split into two feature attributes DI_1 e DI_2, each with 180 samples, serving as input data to the ML algorithms.

3 **Feature Selection:** The feature selection is the process of choosing a subset of variables from a given set [37, 38]. In this case, the DIs represent features characterising the tightening torque. These features are employed as inputs for the subsequent pattern recognition stage, performed here by the K-means.

4 **Pattern Recognition and Clustering:** Pattern recognition and clustering are performed using the K-means unsupervised algorithm to group the data into distinct clusters. In this generated dataset, there is prior knowledge regarding the number of clusters to be chosen (e.g. $K = 5$). However, the elbow method can also be applied to determine the most appropriate number of clusters for K-means. Thus, in this clustering step, the elbow method was applied to the dataset to validate this assumption. The result closely aligned with our assumed value of K=5, reinforcing the appropriateness of our choice. The K-means algorithm receives DI_1 and DI_2 attributes and returns clusters with 180 samples grouped between healthy and damaged. The utilisation of clustering techniques offers a significant advantage in this step. By grouping similar data, it becomes possible to describe the unique characteristics of each group efficiently, giving us clear pattern recognition. This step falls under the data preparation stage. It holds importance as it will serve as a base for ML algorithms in the classification and regression, which will be subsequently applied to the identified clusters.

5 **Data Splitting:** In this step, the new dataset generated by K-means (two attributes and a group cluster) is divided into training and testing sets for model construction and evaluation. The training subset is used for model training, while the test subset is reserved for assessing its performance. Train and test splitting data adopted in work is described in Table 1. One assumes 70% of the data for training and 30% for testing.

Bolt Loosening Detection:

6 **Classification:** This step involves the application of six supervised classification ML algorithms to the training data. This process allows the model to learn patterns and relationships within the dataset.

7 **Model Evaluation:** This step assesses the model's performance by testing it on the previously separated test dataset (as per step 6). During this stage, model hyperparameters can be fine-tuned to improve metrics such

as Accuracy, Precision, Recall, and F1-Score. Additionally, the confusion matrix is examined for insights. Cross-validation is employed to prevent overfitting and promote model generalisation on the training set, with 5-fold cross-validation being utilised in the current study.

Bolt-Loosening Quantification:

6 **Regression:** The SVR-supervised regression machine learning algorithm is applied. The same dataset previously clustered by the K-means model (two attributes and a group cluster) is used as input in this process, addressing the regression problem, which returns the torque estimated values.

7 **Quantification:** This step quantifies the extent of bolt loosening using the SVR algorithm. This step aims to assess the effectiveness of SVR in estimating torque as an indicator of bolt looseness.

8 **Final Decision and Interpretation:** The last step of the proposed algorithm provides information about the torque state based on the classification algorithm and torque quantification from the regression algorithm. The uncertainty quantification on the toque estimation is also provided. Both results are used for interpretation and decision-making on bolts' torque loosening.

Overall, applying unsupervised-supervised machine learning algorithms and evaluations in structural health monitoring can greatly improve the accuracy and efficiency of bolt loosening detection. This semi-supervised approach can provide valuable insights for determining maintenance and repair operations, ultimately leading to improved safety and the longevity of structures.

3 Data-driven processing and pattern recognition

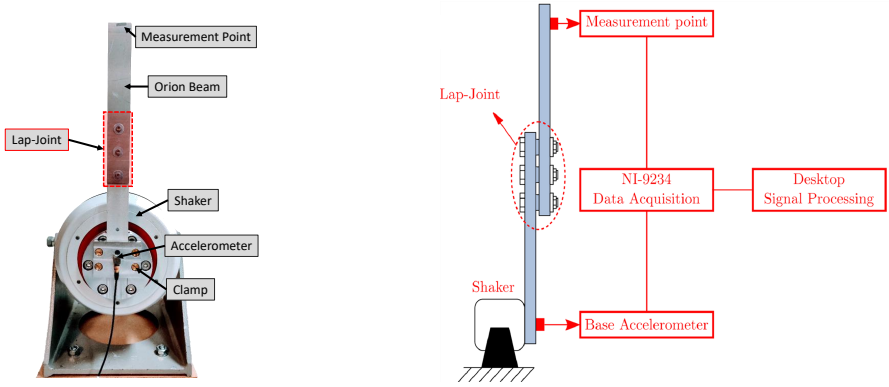
The main objective of the pattern recognition process is to classify objects into different categories or classes based on the analysis of their characteristics [37, 39]. In damage identification, pattern recognition aims to identify structural changes relative to the undamaged state linked to the damaged structural state. This process begins with collecting sensor data from the monitored structure and ends with identifying the damage to assess the actual condition of the structure [40, 41].

3.1 Experimental data-driven acquisition

Bolt loosening detection from vibration data is challenging due to variability and nonlinear effects from the contact interface in bolted joints. Therefore, this paper proposes a data-driven strategy to detect loosening bolts from experimental vibration signals. In the experimental set available in Teloli et al. [36],

the authors consider a physical system consisting of two bolted beams (dimension 370 × 30 × 2 mm) in a cantilever and joint lap configuration connected by three bolts with controlled tightening, as shown in Fig. 2. The experimental apparatus consists of a load cell (PCB288D01), an electromagnetic Modal Shop shaker (Model K2004E01), a 3D scanning laser, NI9234 hardware for data acquisition. The excitation spectrum considered was a white noise Gaussian input with the amplitude levels of $1m/s^2$, $4m/s^2$, $8m/s^2$, and $12m/s^2$ RMS values induced by the shaker at the clamped end. The tightening torques applied on the beam's bolts were 10 cNm, 20 cNm, 30 cNm, and 80 cNm.

The tightening torques were measured with a Lindstorm MA500-1 torque wrench, and the force acquired with a Futek LTH300 donut-load cell was performed after each experimental run. The experimental data-driven consists of the frequency response obtained by dividing the velocity measured at the beam's free edge by the acceleration measured at the clamp. Three-hundred-sixty response samples were acquired considering all different excitation spectrum amplitude levels and variability in assembling the jointed structure.



(a) Experimental setup (b) Schematic diagram

Fig. 2: Physical and schematic drawing of the experimental bolted beams presented in [36].

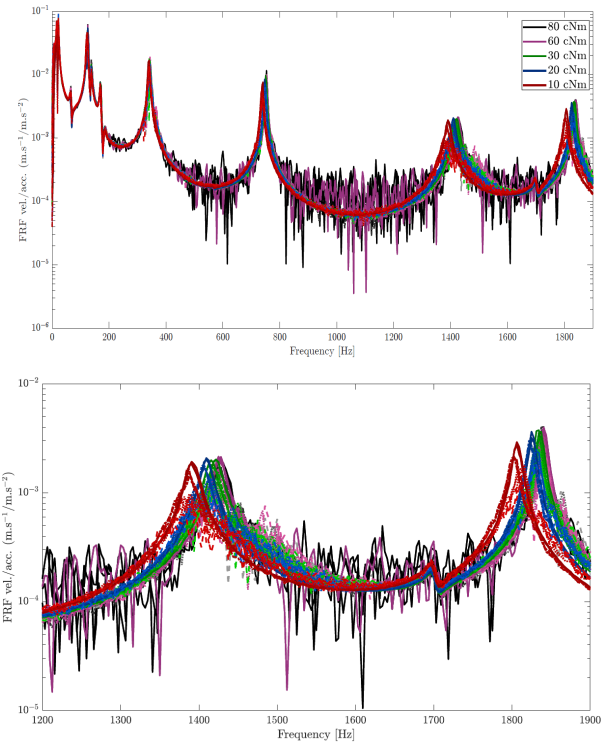


Fig. 3: Twenty samples of the bolted beams data-driven [36]. The entire signal spectrum (on top) and truncated signal comprise the 5th and 6th mode shape (at the bottom). The continuous line represents the based acceleration level of $1 m/s^2$, dashed-dot line acceleration level of $4 m/s^2$, dashed line acceleration level of $8 m/s^2$, and dotted line acceleration level of $12 m/s^2$.

Figure 3 (top) displays four experimental Frequency Response Function (FRF) samples for each torque and base acceleration level measured over a frequency range from 0 to 1900 Hz, in a total of 20 curves printed in the figure. A different colour line represents the FRF of each torque. The continuous line represents a base acceleration level of 1 m/s^2, the dashed-dot line represents an acceleration level of 4 m/s^2, the dashed line represents an acceleration level of 8 m/s^2, and the dotted line represents an acceleration level of 12 m/s^2. The torque loosening induces clearer changes in higher mode shapes, for instance, on the fifth and sixth modes shape, which can be more evident as the excitation amplitude levels increase. There is both noise and variability present in the spectral signal across the entire frequency range. When considering the whole FRF signal in the damage index calculation, the estimation of torque loosening can induce a false positive in the prediction and diagnostic because one considered all signal power densities, including the signal in low frequencies, less influenced by the

damage. Therefore, the truncated signal ranging from 1200 to 1900 Hz was assumed in the DIs estimation (Fig. 3 bottom). This frequency band was the most affected by the torque loosening.

The structure's vibration signature has been used to detect, locate and quantify damage and anomalies in a structure from changes in its dynamic characteristics [50]. Among the methods that employ the dynamic response, the damage index (DI) is a metric that correlates a system signal in different states. The reference signal, usually derived from the system considered an undamaged state, correlated to the one provided by the system under the presence of discontinuing or damage [51]. Various DI approaches have been developed to extract signal features in different domains, aiming at identifying structural damage using an indicator that describes the damage as explored in [30–34]. The DIs are associated with the estimation techniques for damage quantification and reveal important information about the structural health condition. The literature describes a range of DI developed over time. The Frequency Response Assurance Criterion (FRAC) is a damage index representing the correlation between tested frequency responses. It references FRF signals [29], where a unity indicates a strong correlation in case no damage is found. In contrast, the lowest correlation reaches zero, depending on the damage severity. The FRAC is defined by

$$FRAC_{ij} = \frac{\left\| H_{ij}^{d}(\omega) \left(H_{ij}^{u}(\omega) \right)^{*} \right\|^{2}}{\left[H_{ij}^{u}(\omega) \left(H_{ij}^{u}(\omega) \right)^{*} \right] \left[H_{ij}^{d}(\omega) \left(H_{ij}^{d}(\omega) \right)^{*} \right]} \tag{1}$$

where "*" defines the complex conjugate operator, "$H_{ij}^{d}(\omega)$" is the FRF vector on "j" for the damaged excited on "i" and "$H_{ij}^{u}(\omega)$" is the FRF vector for the undamaged, on the same aforementioned coordinates.

The frequency responses are then employed to diagnose and predict the bolt torque loosening. An FRF corresponding to a tightening torque set at 80 cNm of each base acceleration level was assumed to be the reference signal for calculating the DIs with FRAC against other FRFs of respective base acceleration of the bolded beams under undamaged and damaged conditions. Since the experiment involved a specimen assembled and disassembled for each FRF measurement, as described in [36], this procedure introduces variability in the bolted beams' dynamic responses and, consequently, randomness in DIs estimation. In the data acquisition algorithm step, we counted 360 samples encompassing the bolted beam's damaged and undamaged states. The data categorisation follows two attributes, referred to as DI_1 and DI_2, with 180 samples of DI each.

3.2 Feature selection and pattern recognition

The ML algorithm is expected to classify the damage state correctly. In this case, DIs close to unity are considered healthy states of the structure for

torques of 80 and 60 cNm. In contrast, torque losses are associated with torques of 30, 20, and 10 cNm, indicating structural damage. The classification identifies the structure state and quantifies as Healthy(80), Healthy(60), Damage(30), Damage(20), and Damage(10). The DIs are calculated considering the whole signal spectrum, signal truncated comprising 5th and 6th mode shapes, and the signal covering only the fifth and sixth modes separately.

From data processing and estimating DIs, two attribute datasets (DI_1 and DI_2), each consisting of 180 samples, were generated based on the selected FRAC DI observations. Having the dataset pre-processed, DI clustering was carried out using the K-means algorithm with a value of K = 5 (for K representing the number of classes) to identify clusters within unlabelled data. Consequently, an object target dataset was created, comprising 180 samples belonging to one of the 5-labelled classes. Further, the samples were randomly divided, with 70% of the data allocated for training and 30% for testing, as defined in Table 1. Each classifier was assessed using a 5-fold cross-validation procedure to enhance training accuracy, accomplished by randomly partitioning the training dataset into five distinct subsets.

The dataset grouped in classes by the K-means is shown using a scatter plot to characterise the clustered points, enabling us to observe the correlation in two-dimensional space and recognise a pattern of the torque loosening. Figures 4a show the correlation between the two attributes of features (DI_1 and DI_2). The cloud points show an unclear classification of torque loosening, as the points cluster in the range of DIs considered healthy state of bolted beams. Bolt loosening most affects the dynamic response in higher frequency modes; therefore, by using the whole response signal spectrum, the influence of modes in low frequency intends to dominate the signal power spectrum because they have higher amplitudes. In this case, the torque loosening identification through the DI is compromised. Therefore, whether the ML algorithm can correctly classify the damage is unclear. All datasets have a considerable correlation, which can induce misleading classification and false diagnosis identification.

For torques of 80 and 60 cNm the FRAC DI varies between 1 to 0.9, which is considered a healthy structure state. In contrast, torque losses are associated with torques of 30, 20, and 10 cNm, with the DIs estimated from 0.8 to 0, indicating torque loosening. Therefore, by following the DI values, the multiclass dataset can be used to identify the structure state and quantify the severity of the damage from a pattern in the DI values.

Assuming only part of the signal is truncated in the frequency range of the modes' shapes most affected by the torque loosening, the identification turns more consistent. The data correlation is stronger for the sixth mode than the fifth mode shape because of the small change in the frequency response due to the torques from 80 to 30 cNm. Figure 4b shows the data set pattern classification for the signal containing information of the fifth and sixth mode shapes and the correlation between the features attributes. In this case, the

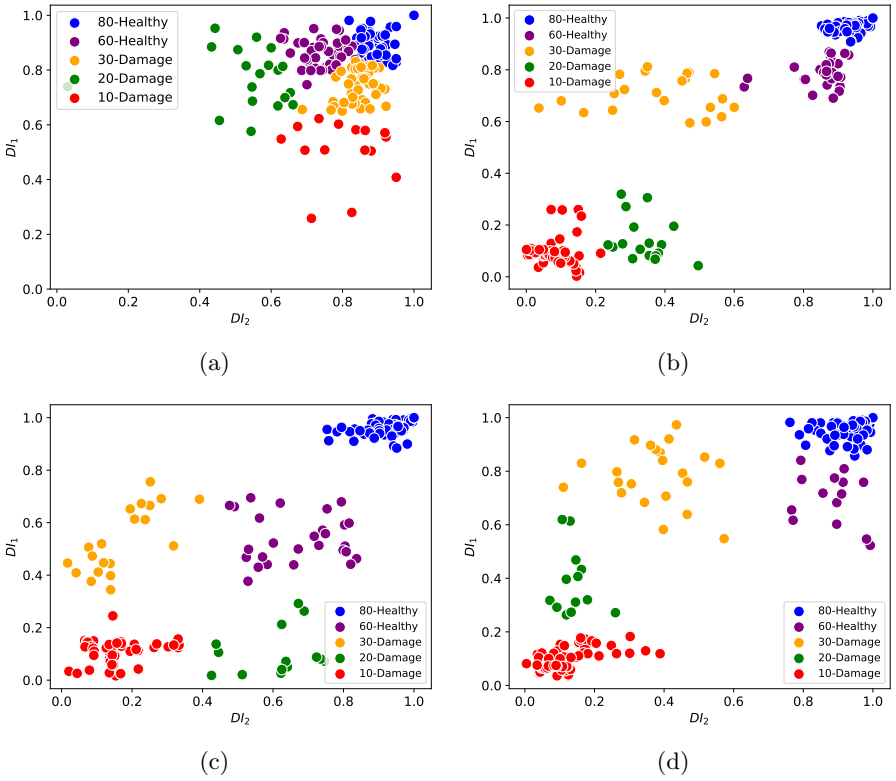


Fig. 4: Correlation scatter plots of between features DI_1 and DI_2 dataset obtained from five different classes of torques in the frequency ranges of (a) 10 to 1940 Hz, (b) 1250 to 1940 Hz (5th∼6th mode), (c) 1250 to 1550 Hz (5th mode), and (d) 1740 to 1940 Hz (6th mode).

classification in healthy and damaged states is more evident as the torque loosening increases. The multiclass dataset can recognise the pattern of the toque loosening through the calculated FRAC DIs. Figures 4c and 4d include information on the mode fifth and sixth, respectively. For each torque level, the DI data points cluster correctly around the range corresponding to the normalised value of DI, indicating that the dataset follows a torque loosening pattern, which can lead to a good classification of the ML algorithm.

The approach for assessing torque loosening based on spectrum signal can effectively capture the intricate dynamics and torque loss processes within bolted joints across varying torque levels. As torque drops, DI consistently decreases in most scenarios. Nevertheless, the FRAC method consistently outperforms others, indicating a clear trend of decreasing values with reduced torque. Therefore, by adopting DI as a normalisation factor in the vibration

Torque (cNm)	Frequency range (Hz)	Classification/Regression					
		Train dataset			Test dataset		
		Healthy	Damaged	Total	Healthy	Damaged	Total
80,60,30, 20,10	1250∼1550 (5th∼6th modes)	60	66	126	26	28	54
	1740∼1940 (5th mode)	69	57		30	24	
	1740∼1940 (6th mode)	59	67		25	29	
Multiclass label classification (torque): Healthy(80), Healthy(60), Damaged(30), Damaged(20), Damaged(10)							

Table 1: Torque values, labelled classification, train, and test data splitting associated with samples number identified healthy or damaged.

dataset, it becomes the input for unsupervised and supervised machine learning algorithms. Consequently, due to DI's normalisation effect on the data, the range of based motion acceleration excitation is dissociated from the subsequent analysis and estimation. In our context, using normalised data is one of the advantages of our algorithm, which lies in the DI procedure. However, a limitation of our proposed algorithm in this paper is that it assumes raw data instead of DI. We are currently exploring ideas to address this issue, but they are beyond the scope of this paper.

4 Machine-Learning techniques for bolt-loosening detection

Bolts have the function of connecting and maintaining stability between two pieces that need to be joined. However, this fixation might only be guaranteed in parts of the structure lifespan, a problem engineering systems face. It is common for the ends joined by the bolts to loosen over time due to external vibrations, dynamic loading, or thermal variations. Bolted joint loosening is damage-like and modifies the connectivity between the components of the structure. Predicting torque losses is essential and helps engineers create control strategies for torque tightening. Based on actual torque data, it is possible to train a classifier model that predicts whether torque loss occurs for a bolted joint and subsequently identifies whether there is damage to the structure.

Machine learning algorithms can be employed in system monitoring using the dataset related to the loss of torque problem. Machine Learning is an automated process that extracts information from data based on a pattern learned through different algorithms. It utilises the learned patterns to predict future data or perform other types of decision-making. This work predicts the state of the bolted connection with unsupervised-supervised learning from the bolted beams' vibration response data-driven. The ML algorithm quantifies the torque

loosening using a regression algorithm under different torque conditions as input. The learning approach is divided into an unsupervised-supervised classification and regression problem, as the data has defined attributes. The ML data input is the DIs, and the target variable will be the information on bolt loosening and indicate the influence of the independent variable. Thus, we intended to obtain a previous classification, determine which unclassified category the data belongs to, and then quantify the torque loosening.

The classifiers algorithm used are the unsupervised K-means to cluster the data and supervised Naive Bayes, Decision Tree (DT), Random Forest (RF), K-Nearest Neighbours (KNN), Support Vector Machine (SVM), and extreme Gradient Boosting (XGBoost) to detect the torque loosening. General theoretical details on the machine learning techniques used here are presented in Appendix A. Each algorithm has hyperparameters that must be configured and tested for optimal performance in application cases. In the case of SVM, a linear kernel function was used, and a grid search was conducted to determine the penalty parameter, assumed as $C = 10$. For KNN, the number of neighbours is set to $k = 3$, and the metric is defined as the Euclidean distance. The function weights are uniform, meaning that all points in each neighbourhood are equally weighted, and the leaf size, which affects query construction and speed, is set to 30. For RF and DT, the number of trees in the forest is 100, and the maximum depth is restricted to 3. The minimum sample split is 2, indicating the minimum number of samples required to split an internal node. The minimum sample leaf represents the training samples on each of the right branches, and the minimum sample leaf values are set to 1. The Max features value is set to 'auto', representing the number of features considered when searching for the best split, and the criterion used is the Gini index. In the case of XGBoost, the objective is assumed as *softmax* for multiclass classification using the softmax objective. The learning rate is set to 0.3, which means that each weight in all trees will be multiplied by this value, and the maximum depth is set to 6. In the Naive Bayes classifier, the Gaussian-NB case was selected. Table 2 shows the hyperparameters selected for each ML algorithm implemented in this paper. All algorithms were applied using the open-source Scikit-learn library in Python.

Algorithm	Hyperparameters
K-Means	Number of clusters = 5; Method for initialisation: k-means++; Number of times the algorithm is run with different centroid seeds = 20.
SVM	Kernel: Linear; C = 10.
K-NN	Metric: Euclidean distance; Number of neighbours: 3.
Naive Bayes	Gaussian
Ramdon Forest	Number of forest trees = 100; Maximum depth = 3; Minimum division = 2; Minimum value of sample sheet = 1; Criterion = Gini Index.
Decision Tree	Criterion = Gini Index; Splitter = 'best'; Maximum depth = 3.
XGBoost	Objective ='multi:softmax', Maximum depth = 6, learning rate = 0.3.

Table 2: Hyperparameters assumed for each ML algorithms.

After the machine learning algorithm completes its estimation, it becomes crucial to assess the stability and accuracy of the model. This validation process involves confirming the quantified relationships between variables, which can be accomplished by examining metrics such as accuracy, score, precision, and recall [47, 48]. However, it's important to note that these metrics primarily reflect the ML model's performance on the data it was trained on. Therefore, the ML model's cross-validation using a separate dataset is necessary to ensure that it successfully captures the underlying patterns in the data, and a reliable validation set indicates a model with low bias or variance. In the damage assessment, the validation and the cross-validation of the ML algorithms are explored. The evaluation metrics of the ML algorithm are addressed to compare the damage detection capability through their accuracy and the confusion matrix. Accuracy close to 100% is considered a good performance. In all cases presented in this paper, the ML model exhibits excellent performance, as evidenced by a standard deviation of 5% in the cross-validations obtained through five different cluster data.

Initially, an unsupervised clustering algorithm K-means is applied to cluster the data, which were divided into five classes. From the samples obtained through the selection of attributes, the algorithms were evaluated by comparing metrics for the classification algorithm in the training and test sets. The idea was to compare the classification metrics, considering the damage index calculated from the FRFs and torques as a characteristic. Therefore, the studies on the dataset investigate the feasibility and accuracy of the six supervising machine learning on the performed classification. The dataset includes five torque identification classes (see Table 1), with two record attributes as input variables or predictors. Loosening torque identification, considered as damage, is performed with 180 samples separated into three different assemblies and divided into training and testing data. The classification method employed here is "one versus one" [49], where one multiclass classification problem turns into ten binary class classification problems. The metrics for multiclass classification are shown in Table 3 for the torque estimation using the 5th∼6th mode shapes, Table 4 only with the 5th mode, and Table 5 with the 6th mode shape. The estimation using the whole frequency band is discarded because of the low accuracy in the DI grouping.

Table 3: Comparison between metrics of experimental test ML algorithms for FRF data (5th∼6th mode).

Performance Metrics	SVM	KNN	NB	RF	DT	XGB
Cross-validation	97,8	97.2	94.4	98.9	98.9	98.9
Accuracy	98.1	98.1	85.1	100	100	90.7
Precision	98.1	98.1	85.1	100	100	90.7
Recall	98.1	98.1	85.1	100	100	90.7
F1-Score	98.1	98.1	85.1	100	100	90.7

Table 4: Comparison between metrics of experimental test ML algorithms for FRF data (5th mode).

Performance Metrics	SVM	KNN	NB	RF	DT	XGB
Cross-validation	98.3	98.9	98.3	99.4	100	99.4
Accuracy	98.1	98.1	100	100	100	100
Precision	98.1	98.1	100	100	100	100
Recall	98.1	98.1	100	100	100	100
F1-Score	98.1	98.1	100	100	100	100

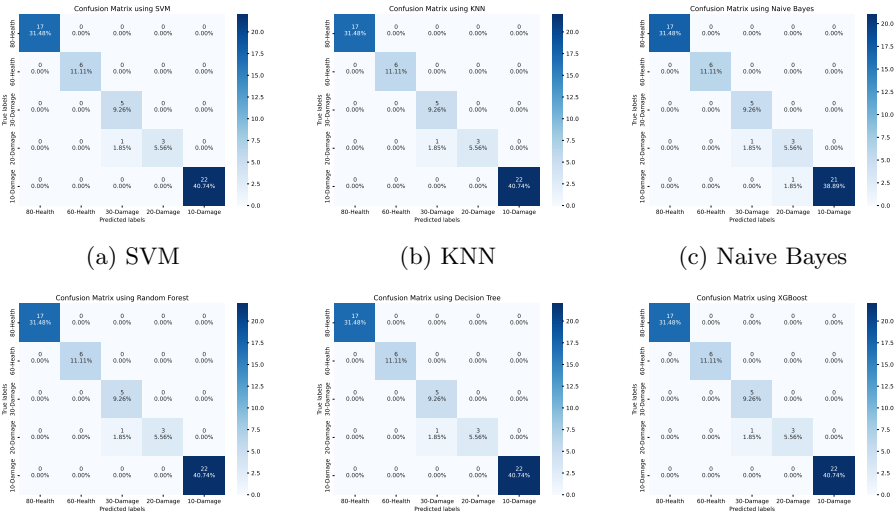
Table 5: Comparison between metrics of experimental test ML algorithms for FRF data (6th mode).

Performance Metrics	SVM	KNN	NB	RF	DT	XGB
Cross-validation	96.7	97.2	97.8	97.8	94.4	95
Accuracy	98.1	98.1	92.6	98.1	94.4	88.9
Precision	98.1	98.1	92.6	98.1	94.4	88.9
Recall	98.1	98.1	92.6	98.1	94.4	88.9
F1-Score	98.1	98.1	92.6	98.1	94.4	88.9

All torque classification performed with the three signals present cross-validation ranging from 94.4% to 100%, and the accuracy, precision, recall and F1-score range from 88.9% to 100%. Torque estimation using the signal from the 5th mode shape presents the higher metrics, showing the efficiency of the algorithms in detecting the most specific conditions for the health and damage state of the bolted beam. The results consistently demonstrate similar performance, yielding accurate predictions on previously overlooked datasets. The XGBoost performed K-folds (K=5) estimation, where 88%, 95.6%, 94.4%, 97% and 100% were achieved. The cross-validation of 95% given by the XGBoost is estimated by the mean value of the five validations previews test. It indicates a high level of consistency in the model's performance across different cross-validation iterations, with minimal fluctuations in its learning behaviour. Cross-validation is a valuable tool for evaluating the algorithm's effectiveness by assessing its performance on various data splits. It provides a robust understanding of how the model behaves regarding overfitting. Additionally, the other metrics support the model's strong classification performance.

The confusion matrix [47] is also vastly employed to verify the data classification, which provides the correct configurations of the classified data. It minimises the error in the damage, and the model's successes provide a comparison between actual and predicted values, where the labels are considered "Positive" and "Negative" [48]. In the case of the torque loss problem, the matrix elements are characterised as true positives (TP), true negatives (TN), false positives (FP), and false negatives (FN). The matrix's main diagonal values show how many correct model predictions are for each class. In this

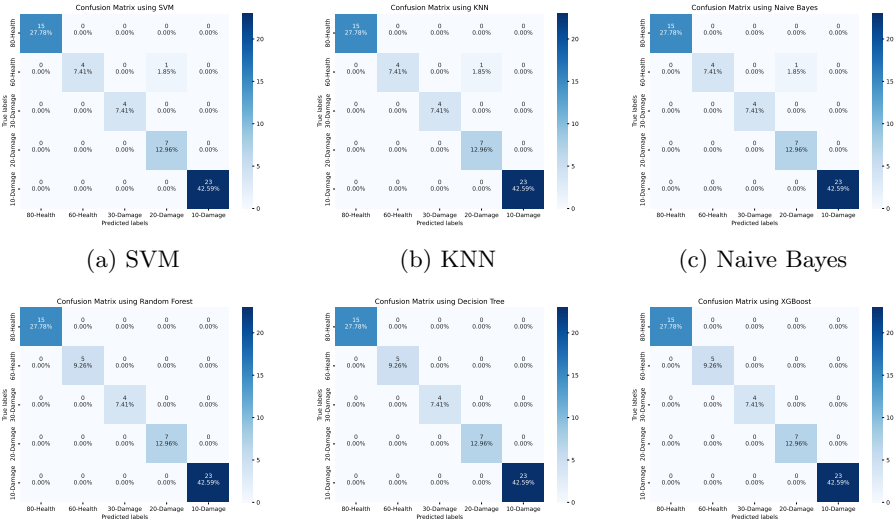
application, the confusion matrix details the ML classifiers' performance in correctly labelling the data and predicting the severity of the loosening torque, which is divided into five classes: damaged or healthy. Therefore, the confusion matrix represents five rows indicating true classes and nine columns representing models' predictions. Figures 5-7 show the confusion matrices of the multiclass classification for torque losses for the truncated signal between the 5th and 6th mode shape. The results indicate that a smaller number of samples were misclassified. Most algorithms had only a sample misclassified, representing 1.85% of the total samples used, in this case, the confusion matrix shows the correct and incorrect predictions on each class. For example, by looking at all the values in row four (Figure 5a), it can be inferred that, out of four samples, the model predicts that three samples belong to class 20-damage (correct prediction) and one sample belongs to class 30-damage. Overall, the result for a multiclass classification using ML algorithms is satisfactory for this specific task using the dataset for bolt loosening identification. For each class, it is possible to evaluate the model's performance by looking at the confusion matrix in detail.



(a) SVM (b) KNN (c) Naive Bayes

(d) Random Forest (e) Decision Tree (f) XGBoost

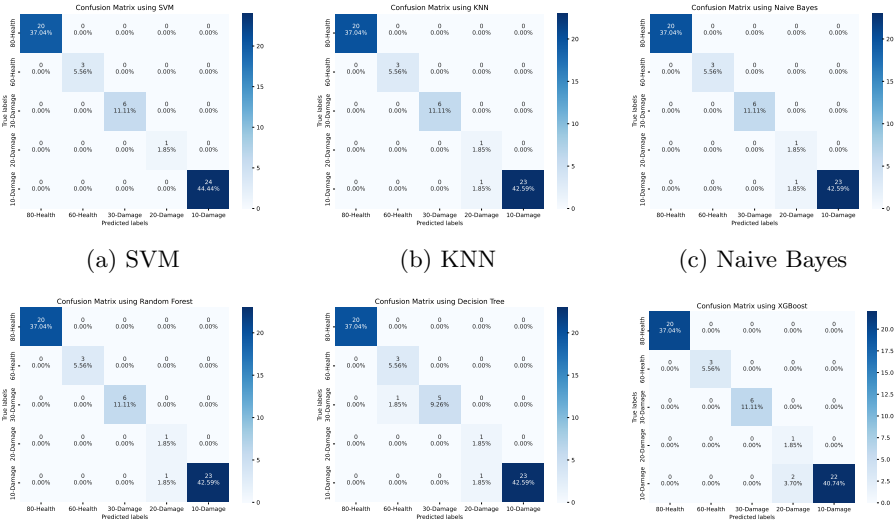
Fig. 5: Confusion matrix of Multiclass classification of six ML techniques in the 5th∼6th mode. The values in blue blocks indicate correctly classified points, whereas those in pale blue blocks indicate misclassified points.



(a) SVM (b) KNN (c) Naive Bayes

(d) Random Forest (e) Decision Tree (f) XGBoost

Fig. 6: Confusion matrix of Multiclass classification of six ML techniques in the 5th mode. The values in blue blocks indicate correctly classified points, whereas those in pale blue blocks indicate misclassified points.



(a) SVM (b) KNN (c) Naive Bayes

(d) Random Forest (e) Decision Tree (f) XGBoost

Fig. 7: Confusion matrix of Multiclass classification of six ML techniques in the 6th mode. The values in blue blocks indicate correctly classified points, whereas those in pale blue blocks indicate misclassified points.

5 Machine-Learning techniques for bolt-loosening quantification

The ML classification algorithm successfully classified torque loosening by identifying the corresponding DI values with good accuracy in the metrics. While the classification algorithm detects the torque loss, the regression method is used to quantify the torque values. Therefore, the regression-based methodology adds to our algorithm the advantage of estimating the torque loss from the input dataset carrying on the intrinsic variability in the data, enabling us to verify the uncertainty associated with the estimation. Therefore, the Support Vector Regression (SVR) algorithm is used to quantify the degree of torque loosening based on score changes. In this process, we employed the same dataset previously clustered by the k-means model to address the classification problem. The regression algorithm also utilised the DI values for torque estimation, with a dataset consisting of two attributes and a group cluster, each one involving 180 samples (given by K-means). It assumes 70% of the data for training and 30% for testing. The SVR algorithm achieved an accuracy score of approximately 98.57%, 98.64%, and 99.17% for the prediction intervals corresponding to the frequency ranges of the 5th to 6th, 5th mode, and 6th mode shapes, respectively.

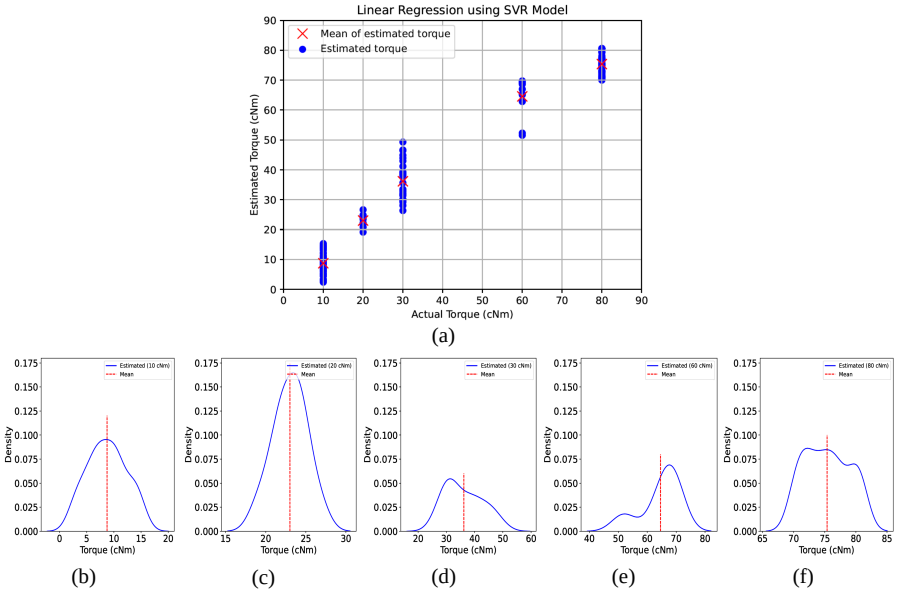


Fig. 8: SVR torque estimation comprises the frequency range comprising 5th and 6th mode shapes in the DI calculation: a) Estimated torque versus actual torque; Probability density function of the estimated torque valuing: b)10 cNm, c)20 cNm, d)30 cNm, e)60 cNm, and e)80 cNm.

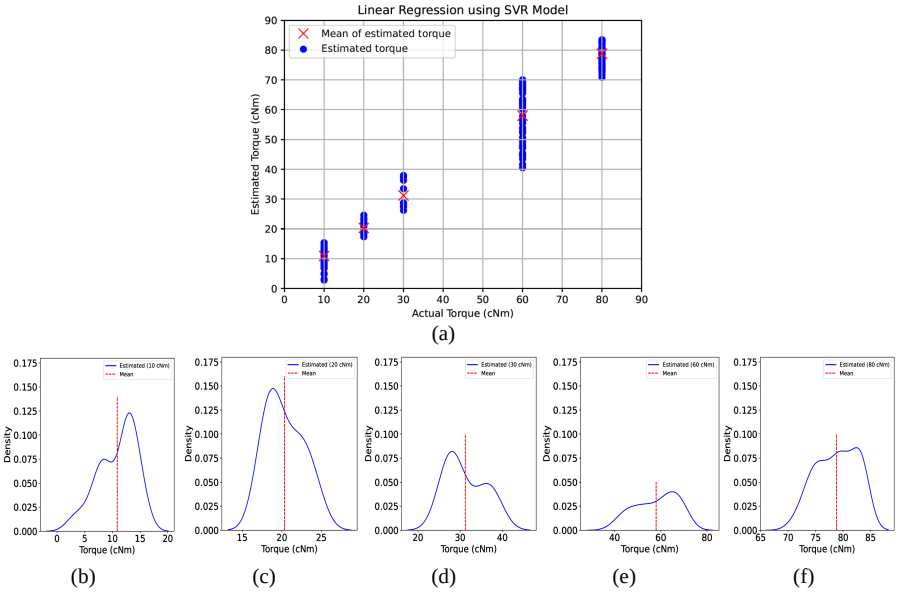


Fig. 9: SVR torque estimation comprises the frequency range comprising the 5th mode shape in the DI calculation: a) Estimated torque versus actual torque; Probability density function of the estimated torque valuing: b)10 cNm, c)20 cNm, d)30 cNm, e)60 cNm, and e)80 cNm.

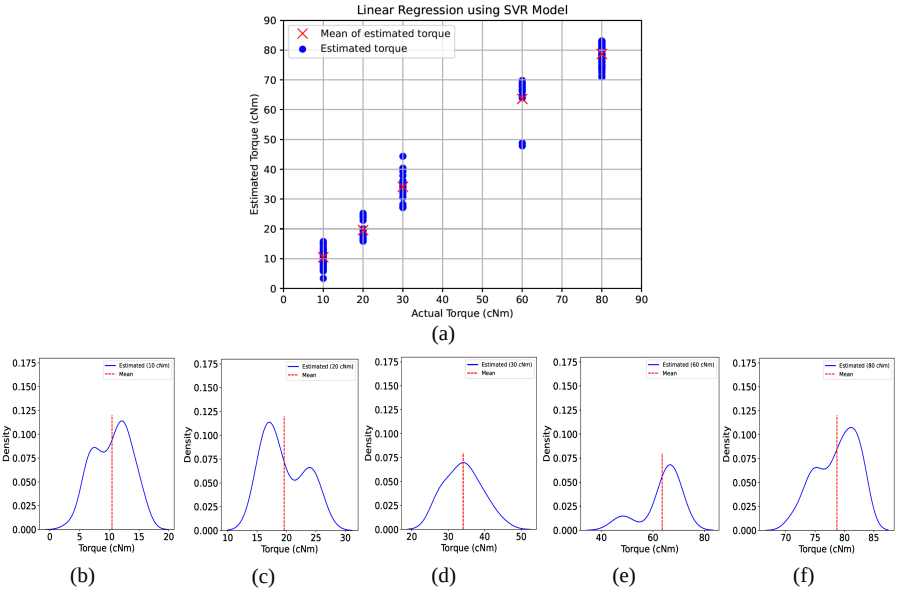


Fig. 10: SVR torque estimation comprises the frequency range comprising the 6th mode shape in the DI calculation: a) Estimated torque versus actual torque; Probability density function of the estimated torque valuing: b)10 cNm, c)20 cNm, d)30 cNm, e)60 cNm, and e)80 cNm.

In Figure 8(a), the plot displays a comparison between the actual torque values and their corresponding estimates. Note that the estimated mean values for each torque condition, indicated by the red 'X' symbols, closely align with the actual torque values. The graphs of Figures 8(b-f) illustrate the probability density functions (PDFs) for estimated torques at various levels: 10cNm ($\mu = 8.73$, $\sigma = 3.52$), 20cNm ($\mu = 23.03$, $\sigma = 2.09$), 30cNm ($\mu = 36.16$, $\sigma = 6.60$), 60cNm ($\mu = 64.56$, $\sigma = 6.62$), and 80cNm ($\mu = 75.37$, $\sigma = 3.52$), respectively. Where μ represents the mean value and σ denotes the standard deviation of the reference torque, calculated using the first and second statistical moments of each PDF.

Figure 9(a) shows the actual torque values versus the estimated torques, with the estimated mean values of each torque. Figures 9(b-f) are the PDF of the estimated torques referent to 10cNm ($\mu = 10.86$, $\sigma = 3.37$), 20cNm ($\mu = 20.32$, $\sigma = 2.35$), 30cNm ($\mu = 31.21$, $\sigma = 4.56$), 60cNm ($\mu = 58$, $\sigma = 8.98$), and 80cNm ($\mu = 78.8$, $\sigma = 3.67$), respectively. In Figures 9, only the 5th mode shape is utilised for the DI calculation, serving as input for the SVR model. In Figures 10(a), it assumed the frequency band of the 6th mode shape for the DI calculation and SVR torque estimation. The SVR successfully obtained all torque values with a good approximation to the actual torque. Figures 10

(b-f) show the PDFs with the mean value indicated in the red line of each estimated torque. In this case, torques estimated with reference values of 10 cNm ($\mu = 10.47$, $\sigma = 3.08$), 20 cNm ($\mu = 19.58$, $\sigma = 3.5$), 30 cNm ($\mu = 34.10$, $\sigma = 4.92$), 60 cNm ($\mu = 63.62$, $\sigma = 7.37$), and 80 cNm ($\mu = 78.7$, $\sigma = 3.48$). Overall, the SVR model provided mean torque predictions with a standard deviation below 10%, which were close to the actual values for most cases. Notwithstanding the foregoing, as stated in Teloli et al. [35], the torque wrench can also show variability in the torque applied - in other words, although the torque value was considered to be a discrete variable, it is a continuous variable, which our method is capable of encompassing.

On this point, the proposed algorithm focuses initially on pattern recognition and loosening torque detection performed with the classification-base ML, and torque estimation is based on the regression. Because of the scattered torque values of the data-driven model adopted in the analysis, the correlation on torque-DI is discontinuous, limiting us from using regression as a linear estimator. In future work, we intend to enrich the experimental dataset and improve the application of the regression in our algorithm.

6 Conclusion

In this study, we investigated the application of six supervised classification machine-learning techniques for detecting bolt loosening and supervised regression techniques for quantifying torques in bolted joints using a vibration-based method. We used data from experimental studies that monitored bolt looseness, considering changes in applied pre-toque and structure assembly and disassembly procedures. We calculated damage indices using the frequency response for the full signal and frequency band gaps ranging from 1220 to 1940 Hz, as the natural frequency in lower modes did not show a significant variation. The FRAC damage index was selected for feature extraction, as it was suitable for generating and pattern-reconditioning the dataset. After feature extraction, an unsupervised algorithm was used to cluster the dataset to discover patterns without human intervention, so the ML algorithms were trained and evaluated using the dataset to determine the state of the bolted connection. The analysis of the algorithms included illustrative examples to show the test error rate through the confusion matrix and metrics.

Based on the test results, the algorithms could classify satisfactorily, with a few classification failures for the case in the three frequency band range conditions. The analysis of the algorithms included illustrative examples to show the test error rate through the confusion matrix and metrics. The performance of the models for classification is above 89% in all frequency band intervals for all metrics. The results obtained through SVR ML demonstrated significant potential in estimating bolt loosening with its uncertainty quantification based on the proposed algorithm methodology, which utilises Damage Indices (DIs) as indicators of damage. Torque estimation achieved with accuracy scores of

98.57%, 98.64%, and 99.17% for the prediction intervals corresponding to the frequency ranges, this approach proves to be an effective tool for predicting damage in structures. In this manner, the applied methodology not only classifies the loss of torque but also provides a good determination of the associated torque value. Overall, this algorithm has shown great potential and accuracy for application in assessing bolt integrity. Therefore, a big challenge in SHM is acquiring and obtaining data, which can be an issue in applying the ML-base detection algorithms.

The application of a linear regression model to estimate statistical moments and model uncertainty may introduce challenges when the data exhibit discontinuities. While this choice aligns with the primary objectives of our study, it is essential to recognise that the model's performance may be affected in scenarios characterised by abrupt changes in the data. Incorporating nonlinear regression models would necessitate a more extensive analysis, which could be pursued in subsequent research endeavours. In future work, we intend to enrich the dataset, improve the application of the regression in our algorithm and improve the algorithm's potential by employing a physic learning algorithm to improve the feasibility of ML algorithms for monitoring continuously bolted joints. This study extensively analyses the classification algorithm but primarily employs linear regression for estimating statistical moments. A comprehensive analysis involving various regression models, including nonlinear ones, can be researched. However, this limitation allows future investigations to investigate the effectiveness of nonlinear regression models in similar contexts.

Acknowledgments. The authors would like to acknowledge the research support of the POLONEZ BIS project SWinT, reg.no. 2022/45/P/ST8/02123 co-funded by the National Science Centre and the European Union Framework Programme for Research and Innovation Horizon 2020 under the Marie Skłodowska-Curie grant agreement no. 945339.

M.R.Machado would like to acknowledge the support from the CNPq Grants no. 404013/2021-0.

Appendix A Machine Learning Techniques

An appendix contains supplementary information that is not an essential part of the text itself but which may help provide a more comprehensive understanding of the research problem, or it is information that is too cumbersome to be included in the body of the paper.

A.1 K-Means Clustering

The K-Means clustering algorithm is an unsupervised ML in which data objects are distributed into a specified number of k clusters [52]. The K is a hyperparameter that specifies the number of clusters that should be created. It is a useful approach for clustering (labelling) or partitioning the data prior

to feeding the labelled data as the output of a supervised ML algorithm. The aim is to find centroids that is a measure of the centre point of the cluster, such that the sum of the squared distances of each data sample to its nearest cluster centre is minimal. Nearest here is with respect to the Euclidean norm (L2 norm). Thus, the objective function is

$$J = \sum_{i=1}^{n} min_k \left(\|x_i - \mu_k\|^2 \right) \tag{A1}$$

where, x_i is referred to as the ith instance in cluster k and μ_k is referred to as the mean of the samples or "centroid" of cluster k.

The K-Means algorithm is widely used due to its simplicity of implementation and low computational complexity but one of the biggest problems of K-Means clustering algorithms include the problem of the initial definition of the number of clusters that must be used. When dealing with highly complex problems where the cluster count is hard to define, the "elbow" method can provide insights into the potential number of required clusters. Another disadvantage of k-means is that it very sensitive to outlier points, which can distort the centroids and the clusters [54].

A.2 K-Nearest-Neighbour Classifier

K-nearest neighbour is one of the simplest supervised learner methods [53, 54] and widely used for pattern recognition[55]. k-NN can be used for classification and regression, where data with discrete labels usually uses classification and data with continuous labels regression. The classification is calculated from a simple majority vote of the nearest neighbours of each point: a query point is assigned the data class with more representatives within the nearest neighbours of the point. For this, a metric between the points is used spaces[54].

The k-NN algorithm, in its simplest version, only considers exactly one nearest neighbour, which is the closest training data point to the point we want to predict. The prediction is then simply the known output for this training point. Depending on the value of 'k', each sample is compared to find similarity or closeness with 'k' surrounding samples. For example, when k = 5, the individual samples compare with the nearest five samples; hence, the unknown sample is classified accordingly [54]. The optimal choice of the value of 'k' is highly data-dependent. In general, a larger suppresses the effects of noise but makes the classification boundaries less distinct.

A.3 Decision Tree and Random Forest

Decision tree supervised algorithm can target categorical variables such as the classification of a damaged or undamaged statement and continuous variables as regression to compare the signal with the healthy state of the system [53]. Learning a decision tree means learning the sequence of if/else questions

that gets us to the true answer most quickly. A tree contains a root node representing the input feature(s) and the internal nodes with significant data information. Each node (also called a leaf or terminal node) represents a question containing the answer. The interactive process is repeated until the last node (leaf node) is reached such that the node becomes impure [54]. The data get into the form of binary features in our application, and a classification procedure is performed.

Random Forest ML algorithm is an ensemble classifier that consists of many decision trees where the class output is the node composed of individual trees. The RF has high prediction accuracy, robust stability, good tolerance of noisy data, and the law of large numbers they do not overfit, and it has been used for structural damage detection. It has shown a better performance [56].

A.4 Support Vector Machine

Support Vector Machines are supervised machine learning techniques developed from the statistical learning theory that can be used for classifying and regressing clustered data. In the case of linear classification, with two classes, let $\{(x_i, y_i), ..., (x_n, y_n)\}$, a training dataset with n observations, where x_i represents the set of input vectors and $y_i(+1, -1)$ is the class label of x_i, the hyperplane is a straight line that separates the two classes with a marginal distance (as seen in Fig. A1). The purpose of an SVM is to construct a hyperplane using a margin, defined as the distance between the hyperplane and the nearest points that lie along the marginal line termed as support vectors [57].

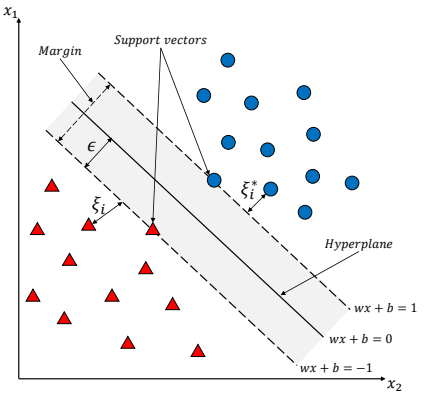


Fig. A1: SVM algorithm operation.

One can define the hyperplane by Eq. (A2), where we have the dot product between x and w added to the term b:

$$D(x) = w^T.x + b = c \quad for \quad -1 < c < 1 \tag{A2}$$

where x represents the points within the hyperplane, w is the weights that determine the orientation of the hyperplane, and b is the bias or displacement of the hyperplane. When $c = 0$, the separating hyperplane is in the middle of the two hyperplanes with $c = 1$ and -1. An SVM aims to maximise the data separation margin from the minimisation of w. This optimisation problem can be obtained as the quadratic programming problem given by

$$min \quad \frac{w^2}{2} \quad s.t \quad y_i(w^T.x_i + b) \geq 1 \quad for \quad i = 1, 2, ..., n \tag{A3}$$

where w is the Euclidean norm.

SVM algorithm encompasses not only linear and nonlinear classification but also linear and nonlinear regression. The main idea of the algorithm consists of fitting as many instances as possible a "tube" while limiting margin violations. Therefore, SVR wants to find a hyperplane that minimises the distance from all data to this hyperplane. The width of the "tube" is controlled by a hyper-parameter, which has an error "insensitive" area, defined by ϵ, as illustrated by Figure A1. The larger the ϵ, the larger the diameter of this tube, and the less sensitive the model is in predicting points within it. In contrast, the smaller ϵ, the smaller the diameter of the tube, the greater the chances of points being on the edges of the tube, making the model more robust. The samples that fall into the ϵ-margin do not incur any loss. Points outside the tube are examined and considered with respect to the ϵ-insensitive region. Compared to a previously defined error, called slack variables (ξ). This approach is similar to the "soft margin" concept in SVM classification, because the slack variables allow regression errors to exist up to the value of ξ and ξ_i^* , yet still satisfy the required conditions. Including slack variables leads to the objective function given by Eq. (A4).

$$\begin{aligned} Minimize: \quad & \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*) \\ Constraints: \quad & \\ & y_i - w^T.x_i - b \leq \epsilon + \xi_i \\ & w^T.x_i + b - y_i \leq \epsilon + \xi_i^* \\ & \xi_i, \xi_i^* \geq 0, \quad i = 1, ..., n \end{aligned} \tag{A4}$$

A.5 Naïve Bayes

Naïve Bayes classification is a probabilistic classification method based on Bayes theorem with the assumption of independence between features, considered a simple technique for constructing classifiers with models that assign class labels to problem instances, represented as vectors of feature values,

where the class labels are drawn from some finite set. There are three classes in sk-learn, the Gaussian-NB, Multinomial-NB, and Bernoulli-NB. The first assumes a Gaussian distribution, the second is for discrete occurrence counters, and the third is for discrete boolean attributes [58].

Naive Bayes classifiers are highly scalable, requiring several parameters linear in the number of variables in a learning problem. Maximum-likelihood training can be done by evaluating a closed-form expression. In other words, one can work with the naive Bayes model without accepting Bayesian probability or using any Bayesian methods. An advantage of naive Bayes is to train a model with few samples [59].

A.6 Extreme Gradient Boosting (XGBoost)

XGBoost (short for Extreme Gradient Boosting) is an efficient implementation of Gradient Boosting Machines (GBM), developed by Tianqi Chen [61], widely recognised for its superior performance in supervised learning. This versatile algorithm is also considered to be an ensemble tree technique that can be used for both regression and classification tasks.

XGBoost follows the concept of weak-learner, where each predictor could be improved by sequentially training new trees to the model [62]. In other words, the XGBoost makes predictions by creating numerous smaller decision trees, also known as subtrees. Each of these subtrees makes predictions for the data, and their individual predictions are combined to form the final prediction for the given input. This ensemble approach helps improve the accuracy and generalisation ability of the predictive model. The process involves iteratively training these subtrees to correct the errors made by the previous subtrees, gradually refining the overall prediction as more trees are added.

Another feature related to XGBoost is that it uses $L1$ and $L2$ regularisation which helps with model generalisation and overfitting reduction. It uses an optimisation strategy that produces better weights as it calculates the weights of the component models. It also uses slightly less tiny component models.

References

[1] Zaman, I., Khalid, A., Manshoor, B., Araby, S. and Ghazali, M. I. The effects of bolted joints on dynamic response of structures. IOP Conference Series: Materials Science and Engineering 50, (2013).

[2] Zhang, M., Shen, Y., Xiao, L. and Qu, W. Application of subharmonic resonance for the detection of bolted joint looseness. Nonlinear Dynamics 88, 1643–1653 (2017).

[3] Miao, R., Shen, R., Zhang, S. and Xue, S. A review of bolt tightening force measurement and loosening detection. Sensors (Switzerland) 20, (2020).

[4] Nikravesh, S. M. Y. and Goudarzi, M. A review paper on looseness detection methods in bolted structures. Latin American Journal of Solids and Structures 14, 2153–2176 (2017).

[5] Huang, J., Liu, J., Gong, H. and Deng, X. A comprehensive review of loosening detection methods for threaded fasteners. Mechanical Systems and Signal Processing 168, 108652 (2022).

[6] Murphy, K. P. Machine Learning: A Probabilistic Perspective. The MIT Press (2012).

[7] Zhou, Y. et al. Percussion-based bolt looseness identification using vibration-guided sound reconstruction. Structural Control and Health Monitoring 29, 2–5 (2022).

[8] Wang, F. and Song, G. 1D-TICapsNet: An audio signal processing algorithm for bolt early looseness detection. Structural Health Monitoring (2020) doi:10.1177/1475921720976989.

[9] Wang F, Song G. Bolt-looseness detection by a new percussion-based method using multifractal analysis and gradient boosting decision tree. Structural Health Monitoring. 2020;19(6):2023-2032. doi:10.1177/1475921720912780

[10] Zhang, Y., Zhao, X., Sun, X., Su, W. and Xue, Z. Bolt loosening detection based on audio classification. Advances in Structural Engineering 22, 2882–2891 (2019).

[11] Kong, Q., Zhu, J., Ho, S. C. M. and Song, G. Tapping and listening: A new approach to bolt looseness monitoring. Smart Materials and Structures 27, (2018).

[12] Tran, D. Q., Kim, J. W., Tola, K. D., Kim, W. and Park, S. Artificial intelligence-based bolt loosening diagnosis using deep learning algorithms for laser ultrasonic wave propagation data. Sensors (Switzerland) 20, 1–25 (2020).

[13] Gong, H., Deng, X., Liu, J. and Huang, J. Quantitative loosening detection of threaded fasteners using vision-based deep learning and geometric imaging theory. Automation in Construction 133, 104009 (2022).

[14] Yu, Y. et al. Detection Method for Bolted Connection Looseness at Small Angles of Timber Structures based on Deep Learning. Sensors 21, 3106 (2021).

[15] Pham, H. C. et al. Bolt-loosening monitoring framework using an image-based deep learning and graphical model. Sensors (Switzerland) 20, 1–19

(2020).

[16] Zhang, Y. et al. Autonomous bolt loosening detection using deep learning. Structural Health Monitoring 19, 105–122 (2020).

[17] Ramana, L., Choi, W. and Cha, Y. J. Fully automated vision-based loosened bolt detection using the Viola–Jones algorithm. Structural Health Monitoring 18, 422–434 (2019).

[18] Zhao, X., Zhang, Y. and Wang, N. Bolt loosening angle detection technology using deep learning. Structural Control and Health Monitoring 26, e2292 (2019).

[19] Cha, Y. J., You, K. and Choi, W. Vision-based detection of loosened bolts using the Hough transform and support vector machines. Automation in Construction 71, 181–188 (2016).

[20] Razi, P., Esmaeel, R. A. and Taheri, F. Improvement of a vibration-based damage detection approach for health monitoring of bolted flange joints in pipelines. Structural Health Monitoring 12, 207–224 (2013).

[21] Ziaja, D. and Nazarko, P. SHM system for anomaly detection of bolted joints in engineering structures. Structures 33, 3877–3884 (2021).

[22] Eraliev, O., Lee, K.-H. and Lee, C.-H. Vibration-Based Loosening Detection of a Multi-Bolt Structure Using Machine Learning Algorithms. Sensors 22, 1210 (2022).

[23] Miguel, L. P., Teloli, R. de O., da Silva, S. and Chevallier, G. Probabilistic machine learning for detection of tightening torque in bolted joints. Structural Health Monitoring 0, 1–16 (2022).

[24] Chen, R. et al. Looseness diagnosis method for connecting bolt of fan foundation based on sensitive mixed-domain features of excitation-response and manifold learning. Neurocomputing 219, 376–388 (2017).

[25] Zhuang, Z., Yu, Y., Liu, Y., Chen, J. and Wang, Z. Ultrasonic signal transmission performance in bolted connections of wood structures under different preloads. Forests 12, (2021).

[26] Wang, F. A novel autonomous strategy for multi-bolt looseness detection using smart glove and Siamese double-path CapsNet. Structural Health Monitoring 0, 1–11 (2021).

[27] Wang, F., Chen, Z. and Song, G. Monitoring of multi-bolt connection looseness using entropy-based active sensing and genetic algorithm-based least square support vector machine. Mechanical Systems and Signal

Processing 136, 106507 (2020).

[28] Zhou, L., Chen, S. X., Ni, Y. Q. and Choy, A. W. H. EMI-GCN: A hybrid model for real-time monitoring of multiple bolt looseness using electromechanical impedance and graph convolutional networks. Smart Materials and Structures 30, (2021).

[29] Saeed, Z., Firrone, C. M. and Berruti, T. M. Joint identification through hybrid models improved by correlations. Journal of Sound and Vibration 494, 115889 (2021).

[30] Zang, C., Friswell, M. I. and Imregun, M. Structural Health Monitoring and Damage Assessment Using Measured FRFs from Multiple Sensors, Part I: The Indicator of Correlation Criteria. Key Engineering Materials 245–246, 131–140 (2003).

[31] Dallali, M., Khalij, L., Conforto, E., Dashti,A., Gautrelet, C., Machado, M.R., De Cursi, E.S. Effect of geometric size deviation induced by machining on the vibration fatigue behavior of Ti-6Al-4V. Fatigue Fract Eng Mater Struct. 2022; 45(6): 1784-1795. doi:10.1111/ffe.13699

[32] Moura, B.B., Machado, M.R., Mukhopadhyay, T., Dey, S.Dynamic and wave propagation analysis of periodic smart beams coupled with resonant shunt circuits: passive property modulation, European Physical Journal: Special Topics, 2022, 231(8), pp. 1415–1431

[33] Moura, B.B., Machado, M.R., Dey, S., Mukhopadhyay, T., Manipulating flexural waves to enhance the broadband vibration mitigation through inducing programmed disorder on smart rainbow metamaterials, Applied Mathematical Modelling, 2024, 125, pp. 650–671.

[34] M.R. Machado and S. Adhikari and J.M.C. Dos Santos. A spectral approach for damage quantification in stochastic dynamic systems. Mechanical Systems and Signal Processing. 2017; 88: 253-273. doi:ymssp.2016.11.018

[35] Teloli, R. de O., Butaud, P., Chevallier, G., and da Silva, S. Good practices for designing and experimental testing of dynamically excited jointed structures: The Orion beam. Mechanical Systems and Signal Processing, vol. 163, p. 108172, 2022, doi: 10.1016/j.ymssp.2021.108172.

[36] Teloli, R. de O., Butaud, P., Chevallier, G., and da Silva, S. Dataset of experimental measurements for the Orion beam structure. Data in Brief, vol. 39, p. 107627, 2021, doi: 10.1016/j.dib.2021.107627.

[37] Webb, A. R. and Copsey, K. D. Statistical Pattern Recognition. A John Wiley & Sons, Ltd., Publication (2011).

[38] Fieguth, P. An Introduction to Pattern Recognition and Machine Learning. Springer Cham (2022). doi:10.1007/978-3-030-95995-1.

[39] Christopher M. Bishop. Pattern Recognition and Machine Learning. Technometrics vol. 49 (Springer, 2006).

[40] Figueiredo, E. and Brownjohn, J. (2022). Three decades of statistical pattern recognition paradigm for SHM of bridges. Structural Health Monitoring, v. 21, n. 6, p. 3018–3054, 2022, doi.org/10.1177/14759217221075241.

[41] Farrar, C. R. and Worden, K. Structural Health Monitoring: A Machine Learning Perspective. John Wiley & Sons Ltd (2013).

[42] Dutkiewicz, M., Machado, M. R. (2019). Measurements in Situ and Spectral Analysis of Wind Flow Effects on Overhead Transmission Lines. Sound and Vibration, 53(4), 161–175.

[43] Machado, M.R., Adhikari, S. and Dos Santos, J.M.C. Spectral element-based method for a one-dimensional damaged structure with distributed random properties. J Braz. Soc. Mech. Sci. Eng. 40, 415 (2018).

[44] Machado, M.R., and Dos Santos, J.M.C. Reliability Analysis of Damaged Beam Spectral Element with Parameter Uncertainties, Shock and Vibration, vol. 2015, Article ID 574846, 12 pages, 2015. https://doi.org/10.1155/2015/574846

[45] Machado, M.R., and Dos Santos, J.M.C., Effect and identification of parametric distributed uncertainties in longitudinal wave propagation, Applied Mathematical Modelling, 98, 498-517,2021,https://doi.org/10.1016/j.apm.2021.05.018.

[46] Sousa, A.A.S.R., Coelho, J. S., Machado, M.R., Dutkiewicz, M. Multiclass Supervised Machine Learning Algorithms Applied to Damage and Assessment Using Beam Dynamic Response. J. Vib. Eng. Technol. (2023). https://doi.org/10.1007/s42417-023-01072-7

[47] Markoulidakis, I., Kopsiaftis, G., Rallis, I. and Georgoulas, I. Multi-Class Confusion Matrix Reduction method and its application on Net Promoter Score classification problem. ACM International Conference Proceeding Series 412–419 (2021) doi:10.1145/3453892.3461323.

[48] Tharwat, A. Classification assessment methods. Applied Computing and Informatics 17, 168–192 (2021).

[49] Galar, M., Fernández, A., Barrenechea, E., Bustince, H. and Herrera, F. An overview of ensemble methods for binary classifiers in multi-class problems: Experimental study on one-vs-one and one-vs-all schemes. Pattern

Recognition 44, 1761–1776 (2011).

[50] Sinou, J. J. (2009). A review of damage detection and health monitoring of mechanical systems from changes in the measurement of linear and non-linear vibrations. Mechanical Vibrations: Measurement, Effects and Control, June, 643–702.

[51] Barreto, L. S. ; Machado, M. R. ; Santos, J. C. ; Moura, B. B. ; Khalij, L. Damage indices evaluation for one-dimensional guided wave-based structural health monitoring. Latin American Journal of Solids and Structures, v. 1, p. 1-10, 2021.

[52] Absalom E. Ezugwu, Abiodun M. Ikotun, Olaide O. Oyelade, Laith Abualigah, Jeffery O. Agushaka, Christopher I. Eke, Andronicus A. Akinyelu,. A comprehensive survey of clustering algorithms: State-of-the-art machine learning applications, taxonomy, challenges, and future research prospects, Engineering Applications of Artificial Intelligence,Volume 110,2022,104743, ISSN 0952-1976, https://doi.org/10.1016/j.engappai.2022.104743.

[53] Malekloo, A., Ozer, E., AlHamaydeh, M., Girolami, M. (2021). Machine learning and structural health monitoring overview with emerging technology and high-dimensional data source highlights. In Structural Health Monitoring (Vol. 0, Issue 0). https://doi.org/10.1177/14759217211036880

[54] Cutler, J., and Dickenson, M. (2020). Introduction to Machine Learning with Python. O'Reilly. https://doi.org/10.1007/978-3-030-36826-5-10

[55] Kurian, B., Liyanapathirana, R. (2020). Machine Learning Techniques for Structural Health Monitoring. In Lecture Notes in Mechanical Engineering. Springer Singapore. https://doi.org/10.1007/978-981-13-8331-1-1

[56] Zhou, Q., Ning, Y., Zhou, Q., Luo, L., and Lei, J. (2013). Structural damage detection method based on random forests and data fusion. Structural Health Monitoring, 12(1), 48–58. https://doi.org/10.1177/1475921712464572

[57] D. A. Otchere, T. O. Arbi Ganat, R. Gholami, and S. Ridha, "Application of supervised machine learning paradigms in the prediction of petroleum reservoir properties: Comparative analysis of ANN and SVM models," Journal of Petroleum Science and Engineering, vol. 200, no. December 2020, 2021, https://doi: 10.1016/j.petrol.2020.108182.

[58] Flach, P. A., Lachiche, N. (2004). Naive Bayesian classification of structured data. Machine Learning, 57(3), 233–269. https://doi.org/10.1023/B:MACH.0000039778.69032.ab

[59] Russell, S., Norvig, P.(2003) [1995]. Artificial Intelligence: A Modern Approach(2nd ed.). Prentice Hall. ISBN978-0137903955.

[60] Mariniello, G., Pastore, T., Menna, C., Festa, P., & Asprone, D. (2021). Structural damage detection and localization using decision tree ensemble and vibration data. Computer-Aided Civil and Infrastructure Engineering, 36(9), 1129–1149.

[61] Chen, T., and Guestrin, C. (2016). XGBoost: A scalable tree boosting system. Cornell University. https://doi.org/10.1145/2939672.2939785.

[62] M. Chen, Q. Liu, S. Chen, Y. Liu, C. -H. Zhang and R. Liu, "XGBoost-Based Algorithm Interpretation and Application on Post-Fault Transient Stability Status Prediction of Power System," in IEEE Access, vol. 7, pp. 13149-13158, 2019, doi: 10.1109/ACCESS.2019.2893448.