Vibration-based Structural Health Monitoring for beams under temperature variation using integrated acceleration, velocity and displacement features

João Carvalho[1], **Jessé Mazzetti**[1], **Tales Boratto**[1], **Heder Bernardino**[1], **Alexandre Cury**[1], **Leonardo Goliatt**[1]

[1] Federal University of Juiz de Fora (UFJF) – Juiz de Fora, MG – Brazil

{joao.farage,jesse.mazzetti,tales.boratto}@estudante.ufjf.br
{heder.bernardino,alexandre.cury,leonardo.goliatt}@ufjf.br

Abstract. *Damage classification is important in structural health monitoring (SHM), yet achieving high reliability remains challenging due to complex, multi-domain vibration signals. This work focuses on feature engineering, combining derived features from the acceleration, velocity, and displacement domains, obtained by integrating accelerometer data from two datasets. Thus, a framework is proposed to classify levels of damage in beams subjected to different temperatures, through a comparative analysis involving four supervised algorithms: Decision Tree, Random Forest, XGBoost, and Logistic Regression. Finally this approach was effective, since XGBoost achieved over 95% performance on key evaluation metrics with feature combinations from the three physical domains.*

1. Introduction

Structural Health Monitoring (SHM) is a field of research dedicated to monitor the integrity of civil, mechanical and similar structures by predicting potential failures, and as a consequence, ensuring safety, reducing maintenance costs, and preventing catastrophic collapses. Data-driven methods are widely used. Specifically, vibration-based methods monitor the dynamic pattern of structures and detect changes in behavior that may indicate potential damage or failure [Boratto et al. 2025, Alves and Cury 2023]. More specifically, there are four possible levels of approach in SHM: (i) identifying anomalies; (ii) quantifying its severity; (iii) locating the damage; and (iv) estimating the structure's remaining useful life [Alves and Cury 2023].

One of the most well known data-driven approaches in SHM is the feature extraction-based approach, which consists of retrieve informations from the signals to feed machine learning (ML) models. For example, [Boratto et al. 2025] explored statistical and spectral features sets to identify damage in walls, by using smartphone accelerometer data using ML models, such as *K-Nearest Neighbors (KNN)*, *Decision Tree (DT)* and *Random Forest (RF)*. [Qiu et al. 2026] used accelerometer and synthetic data from *Finite Elements Method (FEM)* scenarios to feed KNN and recurrent and neural networks to detect damage, location and severity in bridges.

From the perspective of vibration-based data acquisition, accelerometers are commonly used more often because of their sensitivity at higher frequencies [Taylor 2003] and their easy installation and low computational cost [Ma et al. 2023]. However, in

the context of SHM, velocity and/or displacement informations can also be useful. [Sekiya et al. 2016], for example, estimated displacement of bridges in a traffic dynamic, through accelerometer data. This process was done due to the difficulty in using the displacement sensor, because of its necessity of having a fixed reference point. Meanwhile, [Ma et al. 2023] investigated techniques to measure displacement directly, using accelerometers and other sensors. These alternatives were studied to estimate the displacement of building and bridge structures.

In this sense, instead of installing more sensors along the structure, an alternative is using signal processing techniques to integrate the acceleration signal and get an estimative of the velocity and displacement signals. However, these methods must overcome inherent challenges of signal integration process as low-frequency drifts and high frequency attenuation [Ma et al. 2023]. Imperfections on sensors, DC values or incorrect assumed initial values, for example, can lead to drifts in velocity and displacement [Yang et al. 2016, Gilbert et al. 2010, Ma et al. 2023].

In that regard, many approaches are reported in the literature to integrate the acceleration signal. [Yang et al. 2016] proposed a correction with the upper and lower envelopes of the signal, to determine the baseline offsets and avoid drift in integration. [Huang et al. 2014] established a wavelet-based algorithm, which consists in the convolution of the signal with the integral of a wavelet function, thus the first convolution gives the velocity, and the second convolution the displacement. The work of [Zheng et al. 2019] proposes the use of recursive least squares to the baseline correction and, afterward, the application of a recursive high pass filter, to remove low frequency tendency, with the aim of building a real-time integrator.

[Brandt and Brincker 2014], in turn, performed a frequency domain integration approach, which is the one used in this work. First, the DC component is removed and a *high pass filter (HPF)* is applied to the signal before the integration process. After that, the *Fast Fourier Transform (FFT)* is applied to take the frequency domain and the obtained spectrum is then integrated by dividing by $j\omega$, where j is the complex number and ω is the angular frequency, and then the inverse of the Fourier transform is performed to return to the time domain and obtain the integrated signal.

In this manner, this work aims to provide a data-driven approach in the context of SHM, considering a dataset of vibrations in beams with different levels of damage and with different temperatures. The purpose of this process is building a multiclass classifier framework, that considers each damage level as a different class. As such, it is proposed to use ML models, considering supervised learning, specifically, DT, RF, *Logistic Regression (LR)* and *Extreme Gradient Boosting (XGB)*. In addition, it is intended to evaluate the application of velocity and displacement data, using the integration method presented by [Brandt and Brincker 2014], by analyzing the model's performance and the most important features to the classification. Four sets of features were used for this purpose, considering time and frequency domain, based on features widely used in SHM context, as the set used by [Boratto et al. 2025].

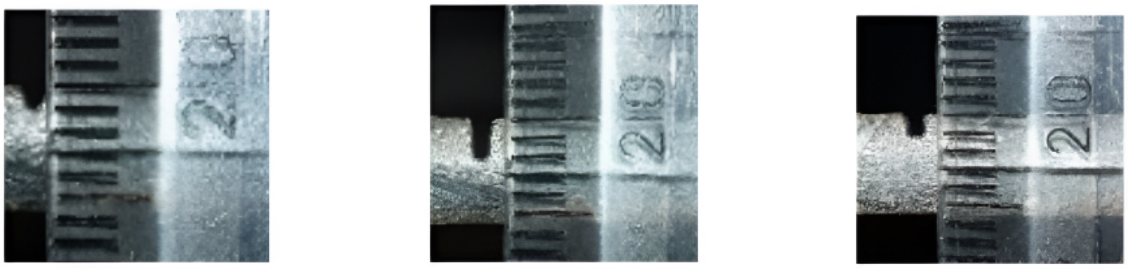
2. Material and Methods

This section presents the dataset we used for this article, as well as the computational models.

2.1. Database

The dataset used in this study was introduced and provided by [Morales et al. 2018]. It corresponds to experimental vibration tests on a steel beam subjected to temperature variation and the introduction of artificial damages, monitored by accelerometers. The beam is installed inside a heating box, which allows the temperature to be raised and controlled throughout the tests. One support is of the first kind and the other of the second kind, allowing the free thermal expansion of the beam as the temperature increases, without inducing significant compressive stresses.

The instrumentation consists of five accelerometers installed along the span, spaced 250 mm apart, connected to a Lynx ADS 1800 data acquisition system operating at 1000 Hz. The beam is excited by forced vibration, through vertical impacts applied with a metallic rod with a rubber tip, positioned 260 mm from one of the supports. Three damage scenarios, as shown by Figure 1 were considered in the beam: in Damage 1 (D1), a 1 mm notch was introduced at midspan; in Damage 2 (D2), the notch at midspan was deepened to 2 mm; in Damage 3 (D3), in addition to the 2 mm notch at midspan, a 1 mm notch was added at the position corresponding to one quarter of the span. These scenarios are combined with the different temperature levels throughout the tests, generating the various structural conditions represented in the dataset.



(a) D1 (b) D2 (c) D3

Figure 1. Damage scenarios.

The tests were carried out at 21 temperature levels, ranging from $19°C$ to $59°C$, with an increment of $2°C$ between consecutive levels. For each temperature, 5 independent vibration samples were acquired and, in each sample, the data from the 5 accelerometers were recorded, resulting in a total of 25 time series per temperature level and damage class (Healthy, D1, D2, and D3). Thus, the dataset is composed of 420 excitation samples (21 temperatures $\times$ 5 repetitions $\times$ 4 damage levels), each one containing the five simultaneous acceleration responses along the beam span, under different combinations of thermal condition and damage scenario.

2.2. Machine Learning Models

Four ML models were used for the experiments: DT, RF, XGBoost, and LR. According to [Ali et al. 2012], the structure of tree-based models (DT, RF and XGBoost) is based on decisions that split the data into increasingly pure subsets to solve the classification. These models stands out for their accuracy and interpretability.

Decision Tree is a ML model widely used for classification and regression tasks due to its simplicity and interpretability. According to [Ali et al. 2012] its structure is

based on a hierarchical organization in which the data are repeatedly divided according to the most relevant features, creating decision rules that guide the prediction process. One of the main advantages of this model is its ease of interpretation, since the generated rules can be directly visualized and understood. In addition, DT can handle both numerical and categorical data without requiring extensive data preprocessing. Furthermore, this model delivers results much faster compared to others. However, despite these advantages, the model is sensitive to overfitting, especially when the tree becomes excessively deep and starts to adapt too closely to the training data, which may reduce its generalization capability.

According to [Ali et al. 2012], random forest is a decision tree-based ML model that combines multiple decision trees to improve predictive performance and model robustness. Instead of relying on a single tree, the algorithm creates several trees using random subsets of training data and features, reducing the dependence on any individual model. The final prediction is obtained by aggregating the outputs of all trees, typically through majority voting for classification tasks or averaging for regression problems. This approach reduces overfitting and increases the model's ability to generalize to unseen data. Moreover, RF usually achieves high accuracy even in complex and noisy datasets, making it one of the most reliable and widely adopted algorithms in practical Machine Learning applications.

Still in the context of tree-based models, [Ahmetoglu and Das 2022] tells that XGBoost is a supervised model based on the gradient boosting technique, in which multiple DT are built sequentially so that each new tree corrects the errors generated by the previous ones. In accordance with [Ibrahem Ahmed Osman et al. 2021] the algorithm is also designed to be computationally efficient, supporting parallel processing, optimized memory usage, and automatic handling of missing values, which makes it highly effective for large and complex datasets. Due to its robustness, scalability, and high accuracy, XGBoost has become one of the most widely adopted algorithms in ML applications.

Finally, [Cramer 2002] emphasizes that LR is a statistical ML model used to predict the probability of a binary event (0/1, yes, no). It estimates the probability of an event and uses the odds ratio to quantify the impact of explanatory variables, that is, it calculates a number n of binary outcomes and performs the operation to find the proportion of these outcomes, thus discovering the probability of an event. Furthermore, the $L1_1$ regularization method was used to avoid overfitting. This regularization method also acts as a tool for feature selection, once it can associate null coefficients to some correlated features.

3. Proposed Approach

The approach used in this article consisted of the implementation of feature engineering and signal processing in the three physical domains (acceleration, velocity and displacement), through the selection of four feature sets. In addition, there is also the hyperparameters optimization.

3.1. Data Processing

The data processing was conducted in 3 stages. First, the DC component was removed and a high-pass filter with a cut-off frequency of 2.5 Hz was applied. This cut-off frequency was selected based on a preliminary analysis of the signals. This data processing

step was adopted to mitigate numerical integration issues, as low-frequency noise, sensor offsets, and baseline errors may produce drift in the estimated velocity and displacement signals. Then, the acceleration signals were integrated in the frequency domain according to the method proposed by [Brandt and Brincker 2014] to obtain estimated velocity and displacement signals. Finally, features were extracted from the three physical domains: acceleration, velocity and displacement.

The feature extraction process was conducted following the strategy proposed by [Boratto et al. 2025], where a set of reference features was selected. After the removal of two attributes, a first set (FS1) was defined, consisting of 19 attributes, as described in Table 1. It should be noted that, for this first set, only acceleration signals were considered. Subsequently, a second set of features (FS2) was proposed, consisting of 16 attributes extracted from the signals in each of the three physical domains. A detailed description of these attributes is presented in Table 2. In addition, a third set (FS3) was defined by combining the two previous sets, ensuring the removal of duplicate features which lead to a set composed of 30 features. Finally, with the aim of eliminating attributes that are highly correlated in absolute terms, a correlation filter was applied to FS3 with a threshold of 0.95 to generate a fourth set of features (FS4) to be evaluated.

Table 1. Description of FS1 features.

Features	Description	Features	Description
X_1	Mean Absolute Value	X_{10}	Spectral Centroid
X_2	Variance	X_{11}	Spectral Decrease
X_3	Root Mean Square	X_{12}	Spectral Distance
X_4	Standard Deviation	X_{13}	Spectral Entropy
X_5	Interquartile Range	X_{14}	Spectral Roll off
X_6	Energy	X_{15}	Spectral Spread
X_7	Mean Absolute Deviation	X_{16}	Max Power Spectrum
X_8	Spectral Skewness	X_{17}	Max Frequency
X_9	Spectral Kurtosis	X_{18}	Median Frequency
		X_{19}	Power Bandwidth

Table 2. Description of FS2 features.

Features	Description	Features	Description
Y_1	Peak	Y_9	Impulse Factor
Y_2	Peak to Peak	Y_{10}	Zero Crossing Rate
Y_3	Root Mean Square	Y_{11}	Activity (Hjorth)
Y_4	Standard Deviation	Y_{12}	Mobility (Hjorth)
Y_5	Skewness	Y_{13}	Complexity (Hjorth)
Y_6	Kurtosis	Y_{14}	Spectral Skewness
Y_7	Crest Factor	Y_{15}	Spectral Kurtosis
Y_8	Shape Factor	Y_{16}	Spectral Centroid

Based on this, it was possible to define two datasets. The first one (dataset I) consists of treating each accelerometer individually as an independent vibration sample, effectively increasing the number of instances. On the other hand, dataset II considers the measurements from all five accelerometers as a single vibration monitoring sample. As a

consequence, the feature sets (FS) associated with Dataset I form a matrix with a fivefold increase in the number of samples (rows), while Dataset II aggregates the features from all five accelerometers into a single unified instance, resulting in a fivefold increase in dimensionality (columns), preserving the original number of temporal observations.

3.2. Hyperparameters Optimization

Adjusting the internal parameters allows the models to be adapted to different scenarios in order to achieve better results, since the goal of this step is to find the most appropriate set of hyperparameters to optimize the classification performance of the problem. Accordingly, this study opted to use grid search combined with Stratified Cross-Validation (SCV).

The SCV was repeated in 30 independent runs. In each run, the data was first split into a training set (75%) and a test set (25%). The training set was then used to perform 5-fold SCV, while the test set was held out exclusively to evaluate the optimized model. The evaluated hyperparameters and the values used in the grid are presented in Table 3.

Table 3. Hyperparameter optimization candidates.

ML Model	Hyperparameter	Candidates
Decision Tree	max_depth	[2, 5, 10, 15, None]
	criterion	[gini, entropy, log_loss]
	splitter	[best, random]
Random Forest	max_depth	[2, 5, 10, None]
	n_estimators	[50, 100, 200]
	criterion	[gini, entropy]
XGBoost	n_estimatores	[50, 100, 200]
	max_depth	[2, 4, 6]
	learning_rate	[0.05, 0.1, 0.2]
Logistic Regression	C	[0.01, 0.1, 1.0, 10.0]

4. Computational Experiments

The computational experiments were conducted in Python following a three-stage workflow. In the first stage, data analysis and feature extraction were performed for all considered scenarios. In the second stage, classification experiments were carried out using 30 independent runs. For each run, the dataset was partitioned to define a holdout test set, while the remaining data were used within a stratified cross-validation (SCV) procedure for hyperparameter optimization. After determining the optimal hyperparameters, the model was retrained on the training data and subsequently evaluated on the predefined test set. In the final stage, the results were consolidated and analyzed, including the distribution of selected hyperparameters, feature importance measures, and overall classification performance.

To evaluate the models' performance, the metrics used were accuracy, precision, recall, F1-Score and ROC-AUC. Furthermore, the confusion matrices were plotted, considering each data set and the model with the best performance, to investigate the quality of the classification. Therefore, the results must be analyzed for each data set and each feature set, assessing the best model for each situation. In addition, the dimension of the

Table 4. Results achieved by each model and feature set (FS) on Dataset I.

FS	# Features	Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
FS1	19	DT	0.7935 ± 0.0173	0.7944 ± 0.0176	0.7935 ± 0.0173	0.7931 ± 0.0174	0.8673 ± 0.0111
		RF	0.8902 ± 0.0142	0.8917 ± 0.0142	0.8902 ± 0.0142	0.8904 ± 0.0141	0.9842 ± 0.0034
		LR	0.8030 ± 0.0131	0.8058 ± 0.0130	0.8031 ± 0.0132	0.8036 ± 0.0130	0.9517 ± 0.0040
		XGB	$\mathbf{0.9126 \pm 0.0127}$	$\mathbf{0.9134 \pm 0.0125}$	$\mathbf{0.9126 \pm 0.0127}$	$\mathbf{0.9126 \pm 0.0126}$	$\mathbf{0.9892 \pm 0.0023}$
FS2	48	DT	0.8208 ± 0.0195	0.8219 ± 0.0199	0.8208 ± 0.0196	0.8206 ± 0.0198	0.8850 ± 0.0138
		RF	0.8971 ± 0.0128	0.8984 ± 0.0128	0.8972 ± 0.0128	0.8973 ± 0.0128	0.9839 ± 0.0033
		LR	0.9022 ± 0.0127	0.9026 ± 0.0125	0.9022 ± 0.0127	0.9020 ± 0.0126	0.9833 ± 0.0037
		XGB	$\mathbf{0.9360 \pm 0.0137}$	$\mathbf{0.9366 \pm 0.0135}$	$\mathbf{0.9360 \pm 0.0137}$	$\mathbf{0.9361 \pm 0.0137}$	$\mathbf{0.9935 \pm 0.0020}$
FS3	90	DT	0.8211 ± 0.0219	0.8221 ± 0.0217	0.8212 ± 0.0219	0.8210 ± 0.0219	0.8863 ± 0.0147
		RF	0.9328 ± 0.0103	0.9338 ± 0.0100	0.9328 ± 0.0103	0.9330 ± 0.0103	0.9926 ± 0.0019
		LR	0.9265 ± 0.0110	0.9271 ± 0.0110	0.9266 ± 0.0110	0.9265 ± 0.0111	0.9906 ± 0.0025
		XGB	$\mathbf{0.9582 \pm 0.0111}$	$\mathbf{0.9587 \pm 0.0110}$	$\mathbf{0.9582 \pm 0.0111}$	$\mathbf{0.9582 \pm 0.0110}$	$\mathbf{0.9969 \pm 0.0013}$
FS4	55	DT	0.8007 ± 0.0192	0.8022 ± 0.0195	0.8007 ± 0.0192	0.8005 ± 0.0192	0.8727 ± 0.0159
		RF	0.9213 ± 0.0127	0.9225 ± 0.0124	0.9213 ± 0.0127	0.9215 ± 0.0127	0.9905 ± 0.0021
		LR	0.9139 ± 0.0122	0.9145 ± 0.0123	0.9139 ± 0.0122	0.9138 ± 0.0123	0.9888 ± 0.0026
		XGB	$\mathbf{0.9491 \pm 0.0078}$	$\mathbf{0.9499 \pm 0.0076}$	$\mathbf{0.9491 \pm 0.0078}$	$\mathbf{0.9491 \pm 0.0078}$	$\mathbf{0.9950 \pm 0.0015}$

datasets was also used for analysis as a tiebreaker criterion in cases of statistically similar performance.

When analyzing the results obtained by the first data set — Table 4 — it is possible to see that XGB achieved the best results for all feature sets evaluated across all performance metrics. For the first feature set (FS1), XGB achieved average values over than 91% in accuracy, precision, recall and F1-Score and over than 98% in terms of average ROC-AUC. For the second feature set (FS2), an improvement in results was observed, with the model achieving average values in performance metrics higher than 93% in terms of accuracy, precision, recall and F1-Score and an average ROC-AUC of 99.35 ± 0.20 %. The best results were obtained when considering the third set of features (FS3), achieving an average accuracy and recall of 0.9582 ± 0.0111, precision of 0.9587 ± 0.0110, F1-Score of 0.9582 ± 0.0110 and ROC-AUC of 0.9969 with a standard deviation of 0.0013. Finally, for the fourth feature set (FS4), it was possible to achieved the second-best results across all metrics: accuracy, recall and F1-score of 0.9491 and standard deviation of 0.0078, precision of 0.9499 and standard deviation of 0.0076, and ROC-AUC of 0.9950 ± 0.0015.

For the second dataset, the obtained results are presented in Table 5. In this case, LR model with the best metrics for all feature sets. In FS1, In FS1, the model achieved a classification performance of almost 91% in terms of accuracy, precision, recall, and F1-score. The ROC-AUC, meanwhile, reached approximately 98%. In FS2, the results improved, with average performance ranging between 94% and 95% for accuracy, precision, recall, and F1-score, and an average ROC-AUC of $99.28\% \pm 0.0044$. The third feature sets (FS3), a slight improvement was observed, with average results exceeding 95% across all metrics and, specifically, an average ROC-AUC of $99.48\% \pm 0.0051$. Finally, FS4 performed worse than FS2 and FS3, outperforming only FS1. Specifically, it achieved a performance of approximately $99.15\% \pm 0.0060$ for the average ROC-AUC and between 92% and 93% for the other metrics.

An ANOVA test was conducted on all performance metrics to determine whether the differences in average scores observed among the best models for each feature set are statistically significant. For Dataset I, statistically significant can be observed among the

Table 5. Results achieved by each model and feature set (FS) on Dataset II.

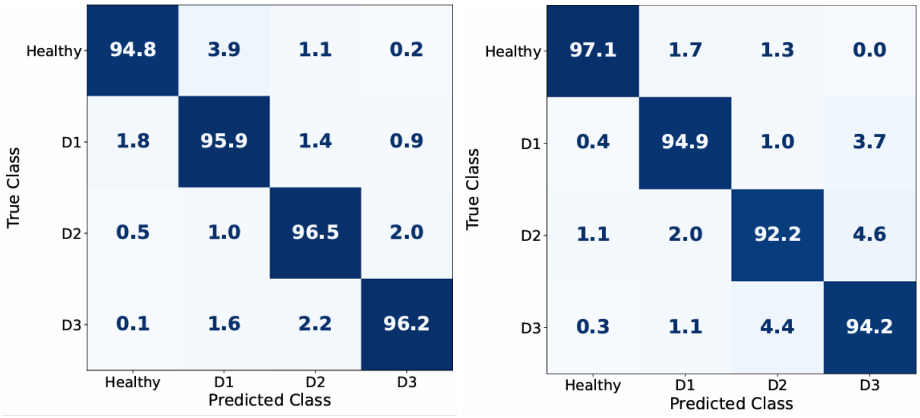
FS	# Features	Model	Accuracy	Precision	Recall	F1-Score	ROC-AUC
FS1	95	DT	0.7867 ± 0.0561	0.7942 ± 0.0551	0.7865 ± 0.0562	0.7849 ± 0.0577	0.8740 ± 0.0300
		RF	0.9022 ± 0.0360	0.9069 ± 0.0330	0.9023 ± 0.0360	0.9022 ± 0.0360	0.9853 ± 0.0100
		LR	$\mathbf{0.9054 \pm 0.0291}$	$\mathbf{0.9099 \pm 0.0270}$	$\mathbf{0.9055 \pm 0.0291}$	$\mathbf{0.9060 \pm 0.0286}$	$\mathbf{0.9841 \pm 0.0078}$
		XGB	0.8965 ± 0.0323	0.9000 ± 0.0312	0.8967 ± 0.0323	0.8963 ± 0.0325	0.9869 ± 0.0081
FS2	240	DT	0.7194 ± 0.0449	0.7298 ± 0.0431	0.7195 ± 0.0450	0.7187 ± 0.0440	0.8336 ± 0.0345
		RF	0.8727 ± 0.0436	0.8776 ± 0.0409	0.8729 ± 0.0437	0.8728 ± 0.0437	0.9759 ± 0.0124
		LR	$\mathbf{0.9460 \pm 0.0220}$	$\mathbf{0.9480 \pm 0.0210}$	$\mathbf{0.9461 \pm 0.0221}$	$\mathbf{0.9460 \pm 0.0221}$	$\mathbf{0.9928 \pm 0.0044}$
		XGB	0.8946 ± 0.0341	0.8978 ± 0.0329	0.8947 ± 0.0342	0.8946 ± 0.0342	0.9861 ± 0.0084
FS3	450	DT	0.7546 ± 0.0501	0.7656 ± 0.0469	0.7549 ± 0.0501	0.7548 ± 0.0498	0.8496 ± 0.0273
		RF	0.8895 ± 0.0418	0.8943 ± 0.0403	0.8896 ± 0.0420	0.8898 ± 0.0418	0.9820 ± 0.0111
		LR	$\mathbf{0.9521 \pm 0.0191}$	$\mathbf{0.9542 \pm 0.0185}$	$\mathbf{0.9521 \pm 0.0192}$	$\mathbf{0.9522 \pm 0.0191}$	$\mathbf{0.9948 \pm 0.0051}$
		XGB	0.9216 ± 0.0241	0.9242 ± 0.0227	0.9216 ± 0.0241	0.9215 ± 0.0241	0.9919 ± 0.0049
FS4	158	DT	0.7505 ± 0.0522	0.7604 ± 0.0489	0.7507 ± 0.0523	0.7503 ± 0.0529	0.8483 ± 0.0382
		RF	0.8940 ± 0.0374	0.8992 ± 0.0340	0.8940 ± 0.0374	0.8944 ± 0.0371	0.9827 ± 0.0085
		LR	$\mathbf{0.9276 \pm 0.0262}$	$\mathbf{0.9315 \pm 0.0256}$	$\mathbf{0.9277 \pm 0.0263}$	$\mathbf{0.9278 \pm 0.0261}$	$\mathbf{0.9915 \pm 0.0060}$
		XGB	0.9067 ± 0.0361	0.9104 ± 0.0342	0.9066 ± 0.0362	0.9067 ± 0.0360	0.9891 ± 0.0063

four feature sets for all metrics, Post-hoc pairwise comparisons using Tukey's HSD test confirmed that all pairs of feature sets differed significantly after adjustment ($p_{\text{adj}} < 0.05$). These results were consistent across accuracy, precision, recall, F1-score, and ROC-AUC. In particular, FS3 significantly outperformed FS1, FS2, and FS4, while FS4 also significantly outperformed FS1 and FS2. For Dataset II, the Tukey HSD analysis revealed a different pattern. FS2 and FS3 did not present statistically significant differences for accuracy, precision, recall, F1-score, or ROC-AUC ($p_{\text{adj}} > 0.05$). Conversely, both FS2 and FS3 significantly outperformed FS1 across all metrics. FS4 also showed significantly better accuracy, precision, recall, and F1-score than FS1, although no statistically significant differences involving FS4 were observed for ROC-AUC. Moreover, FS2 and FS3 significantly outperformed FS4 in accuracy, precision, recall, and F1-score, whereas their ROC-AUC values were statistically comparable. Thus, although FS3 achieved the highest average performance values for Dataset II, FS2 was selected as the preferred feature set because it achieved statistically equivalent performance with substantially fewer attributes.

The confusion matrixes of the best scenarios are shown in Figure 2. Dataset I achieved its highest classification rate for D2 (96.5%), whereas the Healthy class presented the highest confusion rate (3.9%), mainly being misclassified as D1. For Dataset II, Healthy samples were classified mostly accurately (97.10%), while the main confusion occurred between D2 and D3 (4.6%).

4.1. Parametric and Feature Importance Analysis

Following the analysis of the results and the identification of the most promising classification scenarios, an evaluation of the best hyperparameter sets obtained across independent runs, was conducted. Initially, when considering the scenario associated with dataset I (Figure 3), it is observed that a learning rate of 0.2 was the most frequently selected value, occurring in 26 out of 30 runs, thus standing out as the most suitable configuration among the evaluated options. Regarding the maximum tree depth, a value of 4 was selected in 20 out of the 30 runs, and is therefore considered the most appropriate. Additionally, the use of 200 estimators in the XGBoost model was nearly unanimous across



(a) Dataset I — FS3 — XGB **(b) Dataset II — FS2 — LR**

Figure 2. Confusion matrix of the best cases for each dataset.

the experiments. In contrast, for the scenario involving dataset II, the tuning of the single hyperparameter of the Logistic Regression model indicated that the value $C = 10.0$ was the most suitable, being selected in 25 out of 30 runs.

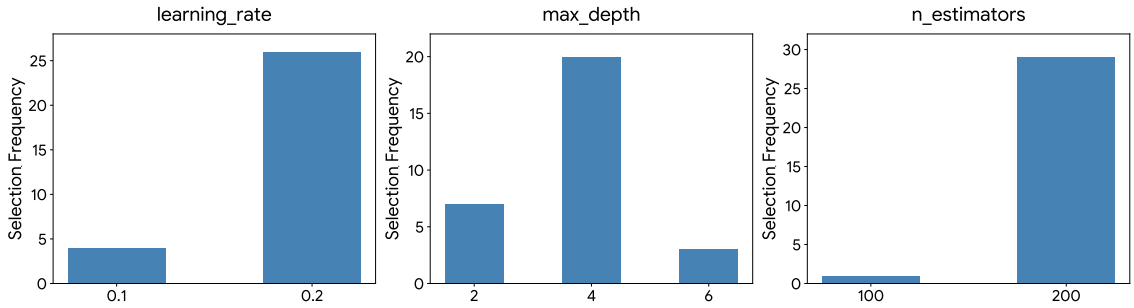


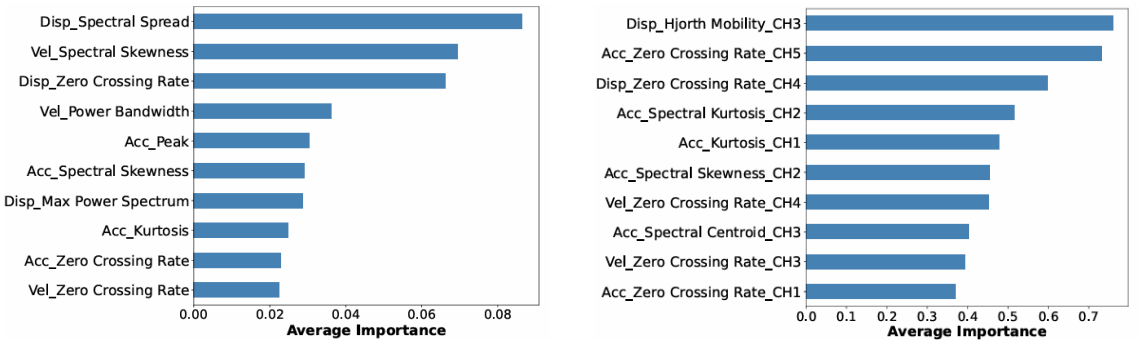
Figure 3. Best hyperparameters found across the independent executions considering XGB combined with FS3 evaluated on DS1.

An evaluation of the most important features for each scenario was also conducted, using the importance scores generated by the tree-based models. For each experimental repetition, feature importance values were computed and subsequently averaged across all runs. The resulting rankings allowed the identification of the vibration descriptors that contributed most significantly to the classification task. For the LR model, this analysis was performed by considering the model coefficients associated with the features.

As shown in Figure 4, when considering the best-performing scenario for dataset I, it can be observed that the most important features are predominantly associated with the velocity and displacement domains, which were introduced in this study. In particular, the features Spectral Spread (Displacement), Spectral Skewness (Velocity), and Zero Crossing Rate (Displacement) achieved the highest importance levels in this scenario. Subsequently, the features Power Bandwidth (Velocity), Peak (Acceleration), Spectral Skewness (Acceleration), and Max Power Spectrum (Displacement) are also identified as relevant, although with comparatively lower impact.

For the scenario involving dataset II, a more balanced contribution across the phys-

ical domains can be observed among the most relevant features. As shown by Figure 4b, it is possible to list the 10 most important features. In descending order of importance, Hjorth Mobility (Displacement) on Channel 3 and Zero Crossing Rate (Acceleration) on Channel 5 were identified as the most influential features, i.e., those with the highest associated coefficients. Following these, the features Zero Crossing Rate (Displacement) on Channel 4, Spectral Kurtosis (Acceleration) on Channel 2, Kurtosis (Acceleration) on Channel 1, Spectral Skewness (Acceleration) on Channel 2 and Zero Crossing Rate (Velocity) on Channel 4 also appear among the most important. Spectral Centroid (Acceleration) and Zero Crossing Rate (Velocity) on Channel 3 presented similar levels and then Zero Crossing Rate (Acceleration) on Channel 1 wraps up the list of the 10 most important features.



(a) Dataset I — FS3 — XGB. **(b) Dataset II — FS2 — LR.**

Figure 4. Feature importances on the best scenarios.

4.2. Overall Discussion

The proposed approach of using features based on the velocity and displacement domains, in addition to the acceleration domain, led to a significant improvement in performance on both datasets. For Dataset I, this improvement was observed across all models. However, for Dataset II, only the LR model outperformed the baseline, which may be related to the higher dimensionality of the feature sets. In this scenario, the use of L1 regularization in LR may have contributed to its robustness, as it promotes sparse solutions by assigning null coefficients to less informative attributes.

The inclusion of velocity-based and displacement-based features proved to be fundamental for the classification task, as several of the most relevant features were associated with these domains. Therefore, the use of velocity and displacement features emerges as a promising approach in the context of SHM, as it contributes to improved performance metrics while also providing the most informative features for the classification process. These findings indicate that the performance improvements are not solely associated with an increase in the number of features, but also with complementary information extracted from the integrated signals.

On the other hand, some limitations remain and motivate future research directions, since generalization for other structures and datasets cannot be ensured by the proposed methodology. Thus, this work can be further evaluated on other benchmark datasets, with data from other structures and different laboratory conditions. Moreover, it

is suggested to evaluate the use of deep learning models acting directly on vibration signals to compare with this approach in terms of performance, reliability, and explainability for decision-making.

5. Conclusion

This paper proposed a vibration based SHM framework to classify different levels of damage in beams subjected to different temperatures. Thus, the use of four different sets of features combined with four ML models was evaluated, considering the feature extraction in acceleration, velocity and displacement. The use of features of velocity and displacement, obtained by integrating accelerometer data, was then assessed.

For dataset I, the third feature set provided the best contribution to the classification task when combined with the XGB model, achieving average performance above 95% across all evaluated metrics. For dataset II, FS2 and FS3 achieved statistically similar performance. However, due to its lower dimensionality, FS2 was selected as the preferred feature set, which, when combined with LR, achieved average performance metrics above 94%. Finally, beyond the improved performance, features derived from velocity and displacement domains consistently appeared among the most important in both scenarios.

Despite the promising results, this study has limitations. The evaluation was conducted using a single controlled laboratory dataset related to a steel beam under different damage scenarios and temperature conditions. Therefore, the generalization of the proposed approach to different structures, damage mechanisms, excitation sources, and operating conditions cannot yet be assumed. In addition, the velocity and displacement signals were estimated through frequency-domain integration of acceleration measurements. Although DC removal and high-pass filtering were adopted to mitigate low-frequency drift, the robustness of this procedure should be further assessed under noisier operational conditions.

To overcome these inherent limitations, future work should evaluate group-aware validation strategies for multi-sensor acquisitions, feature-selection methods to reduce dimensionality and bio-inspired optimization algorithms for hyperparameter tuning. It is also proposed to use additional laboratory and real-world datasets, and evaluate the use of deep learning models operating directly on raw vibration signals, comparing reliability and performance with the proposed approach of this work. With such comparisons, we intend to clarify the trade-off between classification performance, computational cost, and the physical interpretability afforded by feature-based classical machine learning models.

Acknowledgments

The authors acknowledge the support of the funding agencies CNPq (grants 307688/2022-4, 409433/2022-5, 304646/2025-3, 407256/2022-9, 303550/2025-2, 402533/2023-2 and 303982/2022-5), FAPEMIG (grants APQ-02513-22, APQ-04458-23, APQ-00032-24, and BPD-00083-22), and Capes (Finance Code 001).

References

Ahmetoglu, H. and Das, R. (2022). A comprehensive review on detection of cyber-attacks: Data sets, methods, challenges, and future research directions. *Internet of Things*, 20:100615.

Ali, J., Khan, R., Ahmad, N., and Maqsood, I. (2012). Random forests and decision trees. *International Journal of Computer Science Issues(IJCSI)*, 9.

Alves, V. and Cury, A. (2023). An automated vibration-based structural damage localization strategy using filter-type feature selection. *Mechanical Systems and Signal Processing*, 190:110145.

Boratto, T., Lima Fonseca, D., Soares Bernardino, H., Borges, A., Cury, A., and Goliatt, L. (2025). On the use of spectral data from smartphone accelerometer signals and constituent material for the identification of damaged walls. *Rev. Inform. Teór. Apl.*, 32(2):52–63.

Brandt, A. and Brincker, R. (2014). Integrating time signals in frequency domain – comparison with time domain integration. *Measurement*, 58:511–519.

Cramer, J. S. (2002). The origins of logistic regression. Tinbergen Institute Working Paper 2002-119/4, Tinbergen Institute.

Gilbert, H. B., Celik, O., and O'Malley, M. K. (2010). Long-term double integration of acceleration for position sensing and frequency domain system identification. In *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, pages 453–458, Montréal, Canada.

Huang, D., Du, J., Zhang, L., Zhao, D., Deng, L., and Chen, Y. (2014). A wavelet-based algorithm for numerical integration on vibration acceleration measurement data. *Journal of Vibroengineering*, 16(7):3469–3479.

Ibrahem Ahmed Osman, A., Najah Ahmed, A., Chow, M. F., Feng Huang, Y., and El-Shafie, A. (2021). Extreme gradient boosting (xgboost) model to predict the groundwater levels in selangor malaysia. *Ain Shams Engineering Journal*, 12(2):1545–1556.

Ma, Z., Choi, J., and Sohn, H. (2023). Structural displacement sensing techniques for civil infrastructure: A review. *Journal of Infrastructure Intelligence and Resilience*, 2(3):100041.

Morales, F. A. O., Cury, A. A., and Peixoto, R. A. F. (2018). Analysis of thermal and damage effects over structural modal parameters.

Qiu, Y., Ahmed, B., Abueidda, D. W., El-Sekelly, W., de Soto, B. G., Abdoun, T., Ji, H., Qiu, J., and Mobasher, M. E. (2026). Damage identification for bridges using machine learning: Development and application to kw51 bridge. *Digital Engineering*, 10:100103.

Sekiya, H., Kimura, K., and Miki, C. (2016). Technique for determining bridge displacement response using mems accelerometers. *Sensors*, 16(2):257.

Taylor, J. (2003). *The Vibration Analysis Handbook: A Practical Guide for Solving Rotating Machinery Problems*. VCI.

Yang, Y., Zhao, Y., and Kang, D. (2016). Integration on acceleration signals by adjusting with envelopes. *Journal of Measurements in Engineering*, 4(2):117–121.

Zheng, W., Dan, D., Cheng, W., and Xia, Y. (2019). Real-time dynamic displacement monitoring with double integration of acceleration based on recursive least squares method. *Measurement*, 141:460–471.