28th INTERNATIONAL CONGRESS OF MECHANICAL ENGINEERING
COBEM 2025
ABCM Brazilian Society of Mechanical Sciences and Engineering
28th ABCM International Congress of Mechanical Engineering
November 09-11, 2025, Curitiba-PR, Brazil

COB-2025-0841
TABULAR PRIOR-DATA FITTED NETWORK FOR STRUCTURAL DAMAGE PREDICTION

Iago Rhudá Ramos
Ricardo Augusto Baena da Costa
Edo Walfrido de Almeida
Helon Vicente Hultmann Ayala
Pontifícia Universidade Católica do Paraná, R. Imaculada Conceição, 1155, Curitiba
iago.rhuda, edo.almeida, ricardo.baena, helon.ayala@pucpr.br

Abstract. *The goal of the present paper is to evaluate the Tabular Prior-data Fitted Network (TabPFN), a recently proposed machine learning algorithm, for predicting structural damage in mechanical structures using vibration-based features. Vibration-based structural health monitoring (SHM) is crucial for early fault diagnosis, yet its efficacy is often challenged by complex signal signatures that typically demand extensive model tuning or resource-intensive deep learning methods. TabPFN, a pre-trained transformer, offers a significant paradigm shift for SHM by adeptly performing classification tasks on small tabular datasets without requiring any dataset-specific hyperparameter optimization, thereby promising enhanced operational efficiency and high predictive accuracy. This study utilizes an experimental dataset derived from a multi-story laboratory structure subjected to multiple, distinct damage states. Features extracted from vibration signals using Autoregressive (AR) modeling, capturing the dynamic response of the structure, serve as the primary input for TabPFN. The model's predictive performance and key computational characteristics, such as its inference speed and memory footprint, are rigorously benchmarked against several established machine learning classifiers through a standard train-test validation protocol. The results compellingly indicate that TabPFN achieves excellent predictive accuracy, generally outperforming the baseline models in robustly classifying the various structural states, a feat accomplished without the conventional need for laborious, dataset-specific hyperparameter tuning. Conversely, the study observed that TabPFN's inference time was notably longer in the current experimental configuration, a factor partly attributed to the adaptation implemented to support classification across more than ten distinct structural states, thereby extending beyond the standard model's native class handling capacity. This contrasted with the quicker processing times of traditional models, while TabPFN's model size remained moderate. The findings confirm TabPFN's significant potential as an exceptionally accurate and user-friendly tool for structural damage prediction, significantly streamlining the model development lifecycle in SHM. However, the observed inference latency, potentially influenced by the multi-class extension, warrants careful consideration for its deployment in real-time, resource-limited structural health monitoring systems. In the full paper, we will present more results regarding model efficiency, particularly performance and inference timing towards more efficient monitoring with edge computing.*

Keywords: *TabPFN, Machine Learning, Structural Damage Prediction, Principal Components Analysis (PCA), Vibration-based Monitoring, Computational Efficiency*

1. INTRODUCTION

The early diagnosis of structural issues is crucial for preventing failures, especially in vibration-prone environments such as those affected by mechanical oscillations or seismic activity. Data-driven methods have gained popularity for their ability to identify subtle patterns in sensor data, with approaches that perform well on small datasets standing out for their efficiency and accuracy. In this context, this study applies the Tabular Prior-Data Fitted Network (TabPFN), a transformer-based classification model, to the binary classification of vibration data from empirical experiments(1). Pre-trained on millions of synthetic datasets, TabPFN emulates Bayesian classifiers via in-context learning, eliminating the need for fine-tuning or hyperparameter optimization. It has outperformed gradient-boosted trees and classifiers such as XGBoost, LightGBM, and CatBoost on small datasets, while matching or exceeding AutoML frameworks with inference times under one second. Its efficiency arises from being offline-trained to approximate Bayesian inference across synthetic problems modeled with structural causal priors, enabling generalization to new tasks without parameter updates (Hollmann et al., 2023).

The pursuit of efficient models for small tabular data has driven the emergence of foundation models. Hollmann et al. (2025) extended TabPFN to larger datasets (up to 10,000 samples), reinforcing its role as a fast, hyperparameter-free alternative to both traditional and AutoML approaches. Its use of in-context learning with priors favoring simple structural causal models underpins its robustness and relevance to SHM problems, where data is often scarce.

[1] The source code for this study is available at: https://github.com/iago-rhuda/TABULAR-PRIOR-DATA-FITTED-NETWORK-FOR-STRUCTURAL-DAMAGE-PREDICTION.git

Alternative strategies have also been explored. Lara-Abelenda et al. (2025) proposed transforming tabular data into images for transfer learning with CNNs, achieving higher accuracy than TabPFN in some cases, underscoring that while TabPFN is competitive, other paradigms may prevail in specific contexts. AutoML research, such as Karmaker et al. (2021), further emphasizes the importance of automation, though these methods still demand considerable human intervention. Against this backdrop, TabPFN stands out by simplifying modeling for small datasets and aligning well with SHM applications that require speed, efficiency, and low implementation complexity. Moreover, TabPFN can be contextualized to SHM by tailoring synthetic data generation during pretraining to reflect structural behaviors and damage patterns, improving its adaptability to real-world monitoring scenarios.

Prior SHM studies validate this research direction. Regan et al. (2017) demonstrated acoustic-based damage detection in wind turbine blades using supervised ML, establishing a benchmark for feature-engineered approaches. Zonzini et al. (2020) presented an IoT-based SHM architecture integrating smart sensors, standardized communication, and ML, proving effective in real structures. These works highlight the growing role of ML in SHM, while also underscoring challenges such as data scarcity and model complexity.

This paper contributes by investigating TabPFN for vibration-based structural damage detection. By leveraging its ability to deliver accurate classifications without hyperparameter tuning, TabPFN offers a practical alternative to conventional ML, particularly in data-limited scenarios. Preliminary results indicate competitive or superior performance across structural states, supporting the potential of foundation models for SHM while motivating further analysis of inference time, scalability, and deployment constraints.

2. CASE STUDY

The study utilized an experimental setup involving a three-story test structure to investigate Structural Health Monitoring (SHM) methods, as detailed in foundational work by Figueiredo et al. (2009). This structure, made of aluminum plates and columns with bolted joints, was mounted on rails for unidirectional movement (x-direction). To simulate damage and induce nonlinear behavior, an additional central column on the upper story was designed to collide with an adjustable stopper on the floor below, inducing nonlinear behavior through varying impact intensity. For analysis purposes, the study considered 17 distinct structural states that include “Undamaged” and “Damaged” conditions, as specified by Figueiredo et al. (2009). The “Undamaged” cases comprise states 1 to 9, which describe the baseline condition, mass addition at different floors, and stiffness reduction at specific columns. The "Damaged" cases correspond to states 10 to 17, which are defined by the introduction of a gap with different sizes (0.20 mm, 0.15 mm, 0.13 mm, 0.10 mm, 0.05 mm), and in some cases, the combination of the gap with the addition of mass. Data acquisition, consistent with the referenced studies, involved exciting the structure ten times for each of the 17 structural states to capture the variability of the data. Five transducers (channels) were used to collect force and acceleration data: Ch1, a load cell that measures the force applied by the exciter (shaker) at the base; Ch2, an accelerometer located at the base of the structure; Ch3, an accelerometer located on the 1st floor; Ch4, an accelerometer located on the 2nd floor and Ch5, an accelerometer located on the 3rd floor. The accelerometers captured the vibrational response at different levels of the structure, with channel Ch5 (3rd floor) being noted for presenting the largest amplitudes and sharpest peaks, due to the amplification at the top, making it more sensitive to structural changes. The resulting dataset comprises a total of 850 tests (10 tests per structural state × 17 states × 5 accelerometers). For each test (case), 8192 samples were collected for each of the 5 channels. With a sampling time (Ts) of 3.125×10^{-3} seconds, the sampling frequency (Fs) is 320 Hz. Thus, each case has a total duration of 25.6 seconds (8192 samples × 3.125×10^{-3} s/sample). The dataset is structured as a three-dimensional array, where the rows correspond to the temporal samples, the columns to the 5 channels and the depth to the 850 tests.

For the primary feature extraction from these vibrational signals, Autoregressive (AR) models were employed, a methodology extensively analyzed for this dataset and following the feature extraction approach detailed in Contente and Ayala (2021) and building upon foundational work by Figueiredo et al. (2009, 2011). The AR model was chosen for its ability to act as an extractor of damage-sensitive features based on its parameters (AR coefficients). An AR(p) model describes the signal measurement at a given time as a linear combination of previous values of the signal, plus an error term. The order of the model, represented by 'p' or 'na', determines the number of parameters to be estimated. For the accelerometer signals (Ch2 to Ch5), AR models of order 30 (na = 30) were fitted for each case (test) and channel, generating coefficients that capture the dynamics of the system, a methodology consistent with Contente and Ayala (2021) and aligning with investigations for this dataset by Figueiredo et al. (2011). These extracted AR coefficients serve as the primary features used to identify patterns associated with the damage states. The dimension of the AR feature matrix, when combining the coefficients of the accelerometer channels (Ch2 to Ch5), results in 170 cases by 120 features (30 coefficients × 4 channels). To explore the capability of these 120 AR-derived features to distinguish between structural states, parallel coordinate plots were generated. Figures 1 and 2 show all features simultaneously, with lines colored according to the condition of the structure (damaged and undamaged).

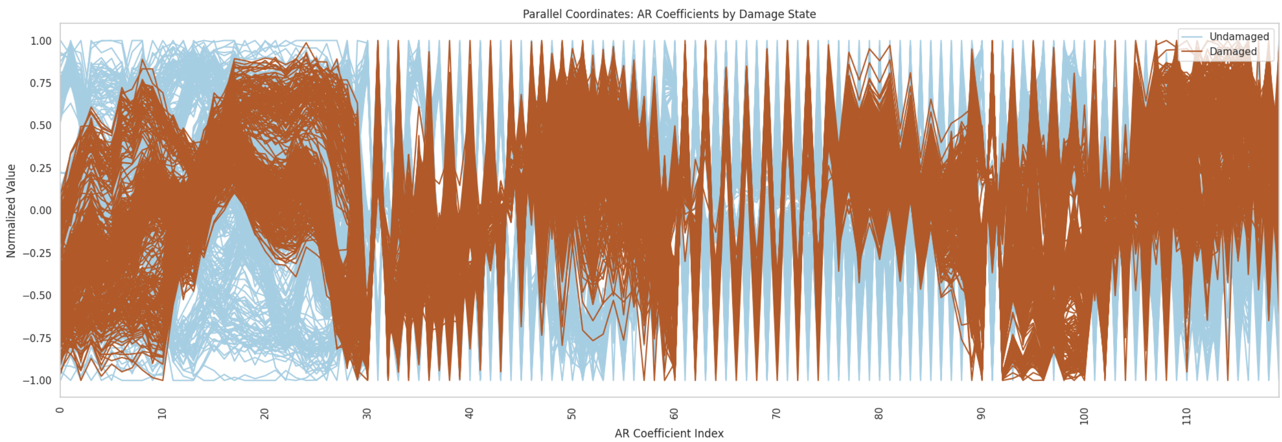

Figure 1. Overlaid Parallel Coordinate Plot - Normalized AR Coefficients (0-110). Light-blue lines (Undamaged) show greater amplitude and variability; brown lines (Damaged) exhibit reduced amplitude (up to 30%) at indices 20-40 and 80-100.

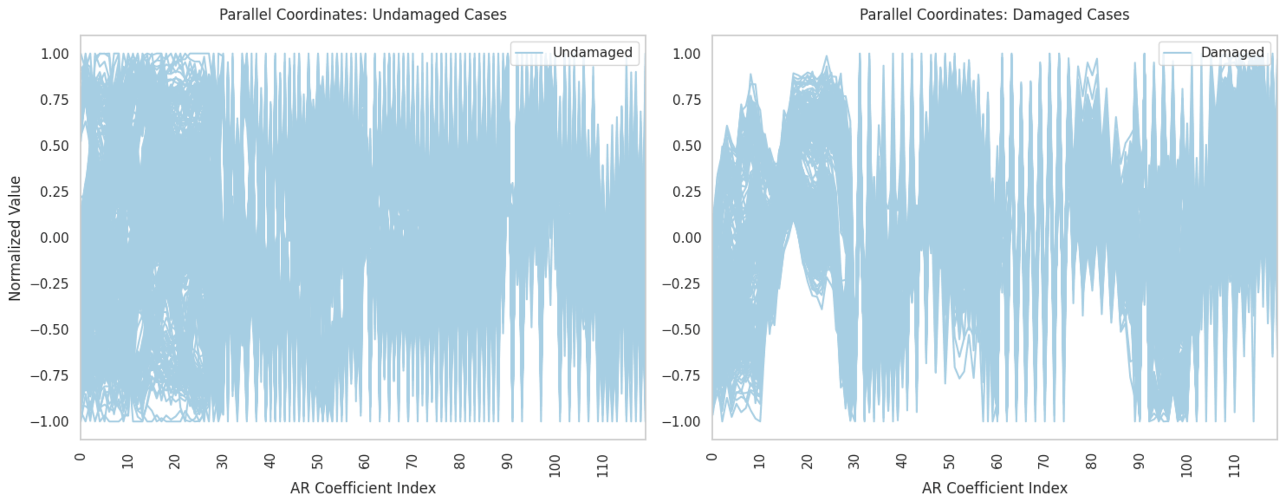

Figure 2. Separated Parallel Coordinate Plots - AR Coefficients by State. Left (Undamaged): wide dispersion (CV ~0.25); Right (Damaged): clustered lines (CV ~0.15), with reduced amplitude at 0-20 and 80-110, suggesting structural failures.

The parallel coordinate plots for the 120 AR coefficients (Figures 1 and 2) illustrate visual patterns that reveal changes in structural dynamics between the "Undamaged" and "Damaged" states. The overlaid plot (Figure 1) displays the normalized values of the AR coefficients by index (0 to 110), highlighting regions where there is a reduction in the amplitude of oscillations in the damaged cases (brown lines) compared to the wider and more diffuse lines of the undamaged cases (light-blue). This reduction, observed in segments such as indices 20-40 and 80-100, is quantified by an average amplitude decrease of 30% (calculated as the difference between the maximum and minimum normalized value per sample), suggesting damage to the structure. This trend reflects the inconsistency in the response data: the undamaged cases exhibit greater variability (coefficient of variation ~0.25), indicating a normal dynamic response, whereas the damaged cases show more clustered trajectories (coefficient of variation ~0.15), with attenuated amplitudes due to a loss of stiffness or changes in vibration. The Ch3 channel, in particular, presents the largest difference, with amplitude variations up to 40% more pronounced between the states. The separated plots (Figure 2) reinforce this distinction: in the undamaged cases (left), the wide dispersion reflects varied but healthy operational patterns; in the damaged cases (right), the concentration of lines highlights the signature of faults, especially in the initial (0-20) and final (80-110) indices. These differences indicate that the AR coefficients capture dynamic changes due to damage, but the large number of features requires methods such as machine learning for robust classification.

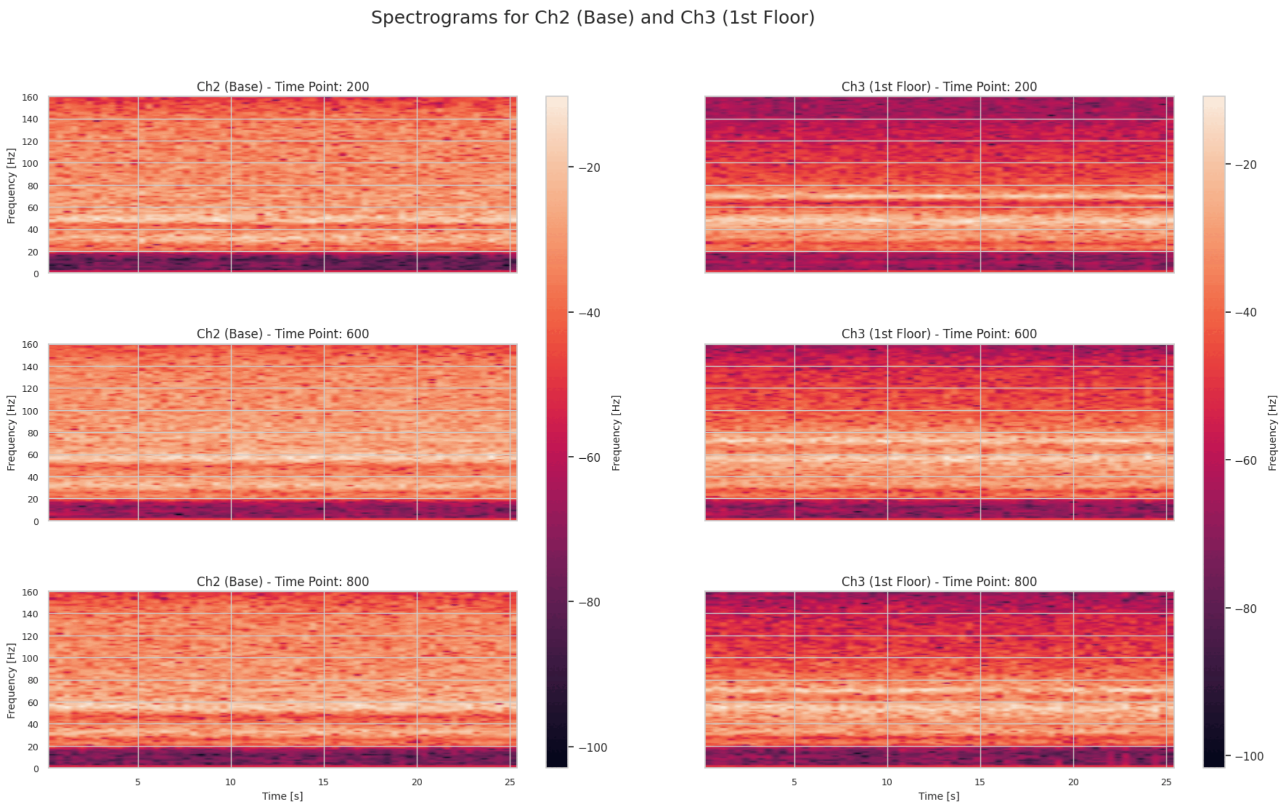

Figure 3. Spectrograms of Channel 2 (Base) and Channel 3 (1st Floor) - Time (0-25.6s) vs. Frequency (0-150 Hz). Ch2 shows average intensity ~10 dB lower; Ch3 shows increased energy (~5 dB) and defined bands, with a variation of 5-10 Hz between states.

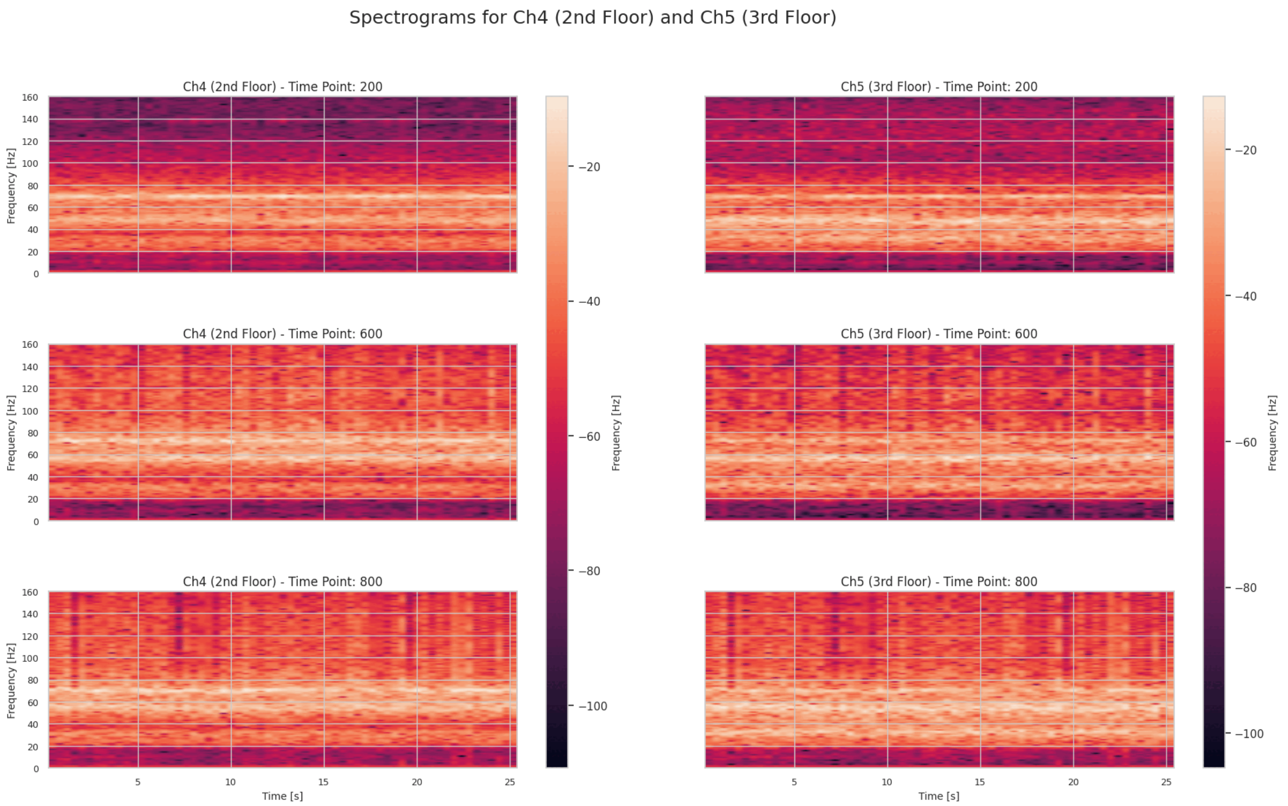

Figure 4. Spectrograms of Channel 4 (2nd Floor) and Channel 5 (3rd Floor) - Time (0-25.6s) vs. Frequency (0-150 Hz). Ch4 with intensity ~15 dB lower than Ch5; Ch5 highlights a 20 dB peak at 50-80 Hz, with a 15 dB reduction and a 10-20 Hz shift in damaged states.

As an alternative exploratory approach, spectrograms were generated from the raw acceleration data of channels Ch2, Ch3, Ch4, and Ch5, as shown in Figures 3 and 4. These spectrograms provide a time-frequency representation of the power spectral density (PSD), allowing for visual inspection of vibrational patterns over the 25.6 seconds of each test, within a frequency range of 0 to 150 Hz (with Fs = 320 Hz). This analysis contrasts with the AR-based method by examining the unprocessed signals directly. The example spectrograms for Channel 2 (Base) and Channel 3 (1st Floor) (Figure 3), and for Channel 4 (2nd Floor) and Channel 5 (3rd Floor) (Figure 4), illustrate the typical distribution of frequency content over the 25.6 seconds, with representative time points (200, 600, and 800). The images show horizontal bands of varying intensity, indicating dominant frequencies and their energy. Channel 2 (Base) exhibits lower overall energy (average intensity ~10 dB below Ch5), reflecting dampened vibrations. Progressively, Channel 3 (1st Floor) and Channel 4 (2nd Floor) display increased energy (a ~5-10 dB increase) and more defined frequency bands, while Channel 5 (3rd Floor) shows the bands with the highest energy and clarity (peak intensity ~20 dB), highlighting the vibrational amplification at the top of the structure and its sensitivity to dynamic changes. When comparing spectrograms across the 17 structural states, visual inspection reveals shifts in dominant frequencies (e.g., a 10-20 Hz shift in Ch5 between undamaged and severely damaged states) and variations in band intensity (a ~15 dB reduction in Ch4 for damaged states), indicating damage. The greatest difference is observed in Channel 5, especially in the 50-80 Hz range, where the intensity significantly decreases in failed states, suggesting a loss of stiffness.

3. METHOD

This section details the methodological framework developed to evaluate the TabPFN for structural damage prediction. The approach involves several critical steps: initial feature extraction from raw vibration signals using Autoregressive (AR) models, subsequent data processing, application and in-depth evaluation of TabPFN as a core classification mechanism, comparative performance analysis with established baseline machine learning models, and finally, the definition of metrics used to quantify predictive accuracy and computational efficiency.

3.1. AR MODELS

In the context of SHM, the autoregressive model was chosen for feature extraction. AR models can be used as damage-sensitive feature extractors based on AR parameters or residual errors. Previous studies have indicated that autoregressive models of order 5, 15, and 30 can represent the behavior of the proposed system satisfactorily for damage detection. In this work, a model of order 30 was constructed. By utilizing the AR parameters from different sensors distributed along the structure of the three-story building, the approach proved to be effective in terms of accuracy when using the most common models. For each channel, the construction of a model of order 30 generates a parameter matrix of 850 x 30.

3.2. TABPFN

TabPFN is a pre-trained AI model designed to solve classification problems on tabular (spreadsheet-like) data very quickly. Unlike other models, it doesn't need to be trained from scratch for each new dataset. Instead, it makes predictions in under a second with a single forward pass. The model was trained just once on thousands of synthetic datasets. This initial training was expensive (20 hours on 8 GPUs) but is a one-time cost. Because of this pre-training, TabPFN uses "in-context learning," meaning it processes both the training and testing data at the same time to make instant predictions. TabPFN is a special type of Transformer model adapted for tabular data. It looks at data both across rows (samples) and down columns (features). This design allows it to work even if the order of the data changes and to handle datasets larger than what it saw during training. Small adjustments, like zero-padding for different numbers of features, make it faster and more flexible. TabPFN excels on small to medium-sized datasets (up to 10,000 samples and 500 features), achieving state-of-the-art results in less than a second. Its performance is competitive with what the best AutoML frameworks achieve in an hour. However, for much larger or highly complex datasets, models like CatBoost, XGBoost, or AutoGluon may perform better. TabPFN is particularly strong on datasets with only numerical data and no missing values.

3.3. BASELINE COMPARISON

The study compared the performance of TabPFN with some established machine learning models, namely RandomForest, an ensemble learning method for classification and regression that operates by constructing a multitude of decision trees during training and generating either the mode of the classes (for classification) or the mean prediction (for regression) of the individual trees; K-Nearest Neighbors (KNN), a nonparametric, instance-based learning algorithm used for classification and regression. In classification, a new data point is assigned to the class of its 'k' nearest neighbors in the feature space; Support Vector Classifier (SVC), a powerful and versatile supervised learning algorithm used for classification. SVC works by finding an optimal hyperplane that best separates data points into different classes in a high-dimensional space; and XGBoost, an optimized distributed gradient boosting library designed to be highly efficient, flexible, and portable. It implements machine learning algorithms under the gradient boosting framework, providing parallel tree boosting.

3.4. EVALUATION METRICS

Model performance was assessed using four key metrics: accuracy, average inference time per prediction, total inference time, and model size. Accuracy measured the proportion of correctly classified instances. Inference times provided insights into computational efficiency, which is crucial for real-time Structural Health Monitoring. Model size, measured in bytes, indicated memory consumption, relevant for deployment in resource-constrained environments.

4. RESULTS

The study compared TabPFN with four traditional classifiers: Random Forest, XGBoost, Support Vector Classifier (SVC), and K-Nearest Neighbors (KNN). Tables 1, 2 and 3 summarize the results for predictive accuracy, inference times, and model sizes.

Table 1. Model performance comparison (MAE) .

Models	Accuracy	Mean	Median	Standard Deviation
TabPFN(1)	0.9918	0.0082	0.0071	0.00470
Random Forest	0.9867	0.0133	0.0118	0.0033
SVC	0.9893	0.0107	0.0094	0.0046
XGBoost	0.9616	0.1264	0.1035	0.0788
KNN	0.9790	0.0210	0.0188	0.0073

(1) no hyperparameter tunning

Table 2. Model performance comparison (inference time).

Models	Average Inference Time (ms) (2)	Median (ms)	Std. Dev. (ms)	Total Inference Time (s) (3)
TabPFN(1)	53.3062	53.4759	1.7580	339.8273
Random Forest	0.0301	0.0300	0.0098	0.1917
SVC	0.0169	0.0122	0.0098	0.1077
XGBoost	0.0182	0.0178	0.0028	0.1163
KNN	0.0192	0.0098	0.0172	0.1224

(1) no hyperparameter tunning
(2) average inference time per prediction (calculated per batch and divided by the batch size)
(3) total inference time per batch

Table 3. Model performance comparison (model size) .

Models	Average Size (bytes)	Median (bytes)	Std. Dev. (bytes)
TabPFN(1)	465,674	465,674	0
Random Forest	1,024,082	1,084,423	349,601
SVC	187,947	163,414	46,019
XGBoost	1,613,991	1,509,694	470,919
KNN	415,319	415,320	0

(1) no hyperparameter tunning

TabPFN achieved the highest accuracy (99.18%), outperforming all baseline models despite the absence of hyperparameter tuning. This result highlights its robustness in classifying the 17 structural states. By contrast, XGBoost showed the lowest accuracy (96.16%) in this task. The main drawback of TabPFN was its high inference time (≈53 ms per prediction), which was orders of magnitude slower than the traditional models (<0.04 ms). In terms of memory footprint, TabPFN was smaller than Random Forest and XGBoost but larger than SVC. Overall, TabPFN combines high predictive accuracy and relatively compact model size, but at the cost of slower inference. These characteristics make it particularly attractive for scenarios where accuracy is prioritized over speed, while real-time applications may still favor traditional methods.. Figure 5 presents detailed confusion matrices for each model, enabling a granular analysis of performance per class and crucial identification of potential biases or confusion between different states, which is essential for diagnostic confidence and decision-making.

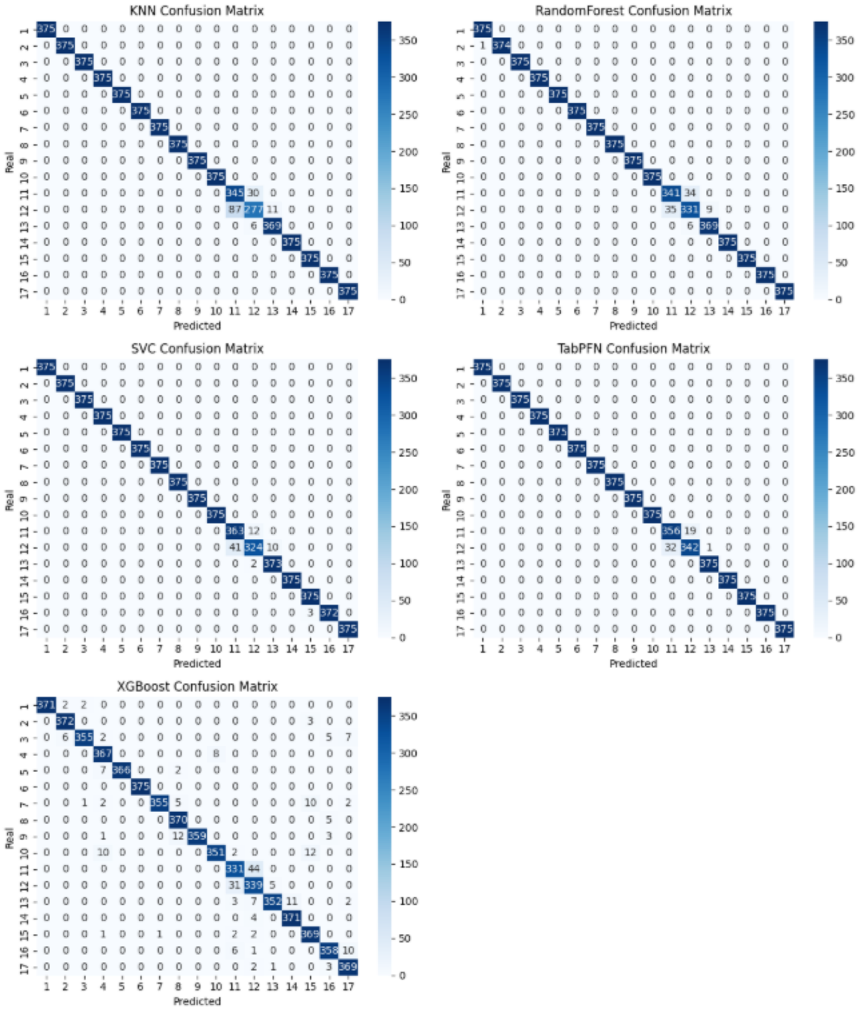

Figure 5. Confusion matrices for the evaluated models, comparing Actual (y-axis) and Predicted (x-axis) classes.

The applicability of machine learning models in embedded or distributed SHM environments depends strongly on inference time and model size. While traditional algorithms such as KNN, Random Forest, SVC, and XGBoost achieved inference times in the order of milliseconds, TabPFN required an average of 53 ms per prediction—still relatively small in absolute terms but several orders of magnitude slower than the baselines. This result contrasts with previous reports in the literature that describe TabPFN as capable of solving small tabular classification problems in less than a second, or even milliseconds with GPU acceleration. The discrepancy is likely due to a combination of hardware limitations and the higher complexity of the SHM dataset, which involves 17 structural states compared to the ≤10 classes and ~100 features typically used in TabPFN's pre-training. Model size also emerges as an important factor for deployment: SVC produced the smallest footprint (≈188 kB), offering a favorable combination of compactness and accuracy, whereas TabPFN (≈466 kB) occupied a mid-range position, smaller than Random Forest and XGBoost but larger than SVC. Unlike traditional models, whose size can often be reduced with techniques such as PCA, TabPFN's architecture is fixed by its pre-trained Transformer core, limiting opportunities for dimensionality reduction.

A more granular perspective is provided by the confusion matrices (Figure 5), which reveal where models struggled with class separation. TabPFN achieved the cleanest diagonal with almost no misclassifications, its few errors appearing in the mid-range classes (12–14). SVC also performed consistently well, though it occasionally confused states 13 and 14. KNN, despite relatively strong overall accuracy, suffered from frequent misclassifications between states 11 and 12, reflecting its sensitivity to local feature overlap. Random Forest maintained a robust diagonal but showed scattered errors across states 10–14, suggesting difficulty with subtle distinctions in vibration patterns. XGBoost was the most affected, with systematic misclassifications among states 11–14 and spillover into state 17, which explains its lower accuracy (96.16%). Taken together, these results highlight the trade-offs: TabPFN delivers state-of-the-art accuracy and highly uniform per-class performance but at the cost of slower inference, SVC strikes the most attractive balance of speed, accuracy, and compactness for real-time embedded SHM, while KNN, Random Forest, and especially XGBoost showed specific weaknesses in distinguishing structurally similar states.

5. CONCLUSION

This study evaluated the use of TabPFN for predicting structural damage in a 17-state mechanical system using vibration features from autoregressive models. Compared with established classifiers such as Random Forest, XGBoost, SVC, and KNN, TabPFN achieved the best predictive accuracy (99.18%), standing out for delivering strong performance without the need for hyperparameter tuning. This makes it attractive for fast deployment, especially in cases where tabular data is limited, as often occurs in structural health monitoring. However, TabPFN also showed higher computational demands: the average inference time per prediction was 53.3 ms, resulting in 339.8 seconds for batch processing—much slower than traditional methods. This slower performance, unlike the near real-time results reported in the literature, may reflect the dataset's complexity (120 features, 17 classes), hardware limitations, or the use of the tabpfn_extensions.many_class implementation, which was necessary to extend TabPFN's default capacity beyond ten classes.

Model size further impacts deployment. TabPFN occupied about 466 kB, larger than SVC (≈188 kB), which also achieved high accuracy and may be more suitable for edge devices with tight resource limits. Unlike conventional models that can sometimes be reduced with techniques like PCA, TabPFN's size is fixed by its pre-trained Transformer core. Overall, TabPFN shows great potential thanks to its accuracy and ease of use, but its slower inference and less flexible footprint are barriers to real-time applications. Future work should explore hardware acceleration, optimized implementations, or compression methods to make TabPFN more practical for edge-based SHM systems.

6. REFERENCES

Contente, C.O. and Ayala, H.V.H., 2021. "Establishing compromise between model accuracy and hardware use for distributed structural health monitoring". In Proceedings of the XV Simpósio Brasileiro de Automação Inteligente - SBAI 2021. pp. 1512-1519.

Figueiredo, E., Figueiras, J., Park, G., Farrar, C.R. and Worden, K., 2011. "Influence of the Autoregressive Model Order on Damage Detection". Computer-Aided Civil and Infrastructure Engineering, Vol. 26, pp. 225-238.

Figueiredo, E., Park, G., Figueiras, J., Farrar, C. and Worden, K., 2009. "Structural Health Monitoring Algorithm Comparisons Using Standard Data Sets". Los Alamos National Laboratory report: LA-14393.

Hollmann, N., Müller, S., Eggensperger, K. and Hutter, F., 2023. "TabPFN: A Transformer That Solves Small Tabular Classification Problems in abh Second". In Proceedings of the International Conference on Learning Representations (ICLR 2023).

Hollmann, N., Müller, S., Purucker, L., Krishnakumar, A., Körfer, M., Hoo, S.B., Schirrmeister, R.T. and Hutter, F., 2025. "Accurate predictions on small data with a tabular foundation model". Nature, Vol. 637, pp. 319-326.

Karmaker Santu, S.K., Hassan, M.M., Smith, M.J., Xu, L., Zhai, C. and Veeramachaneni, K., 2022. "AutoML to date and beyond: Challenges and opportunities". ACM Computing Surveys, Vol. 54, No. 8, pp. 175:1-175:36.

Lara-Abelenda, F.J., Chushig-Muzo, D., Peiro-Corbacho, P., Gómez-Martínez, V., Wägner, A.M., Granja, C. and Soguero-Ruiz, C., 2025. "Transfer learning for a tabular-to-image approach: A case study for cardiovascular disease prediction". Journal of Biomedical Informatics, Vol. 165, p. 104821.

Regan, T., Beale, C. and Inalpolat, M., 2017. "Wind Turbine Blade Damage Detection Using Supervised Machine Learning Algorithms". Journal of Vibration and Acoustics, Vol. 139, No. 6, pp. 061010-1–061010-14.

Zonzini, F., Aguzzi, C., Gigli, L., Sciullo, L., Testoni, N., De Marchi, L., Di Felice, M., Salmon Cinotti, T., Mennuti, C. and Marzani, A., 2020. "Structural Health Monitoring and Prognostic of Industrial Plants and Civil Structures: A Sensor to Cloud Architecture". IEEE Instrumentation & Measurement Magazine, Vol. 23, No. 6, pp. 21-27.

7. RESPONSIBILITY NOTICE

The authors are solely responsible for the content published in this article.