IEEE Access
Multidisciplinary ⁝ Rapid Review ⁝ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2024.Doi Number

Benchmarking Machine Learning Algorithms for Bearing Fault Classification Using Vibration Data: A Deployment-Oriented Study

Prasanta Kumar Samal[1, 2], Pramod Kumar Malik[3], Manjunatha H J[1], Imaran M Jamadar[1], Srinidhi R[2]

[1]Department of Mechanical Engineering, The National Institute of Engineering, Mysuru, Karnataka 570008, INDIA
[2]Department of Mechanical Engineering, JSS Science and Technology University, Mysuru, Karnataka 570006, INDIA
[3]School of Aerospace Engineering, KIIT Deemed to be University, Bhubaneswar, Odisha, 751024, INDIA

Corresponding author: Pramod Kumar Malik (e-mail: pramod.mallickfme@kiit.ac.in).

ABSTRACT This study presents a comprehensive benchmarking of 33 machine learning (ML) algorithms for bearing fault classification using vibration data, with a focus on real-world deployment in condition monitoring systems. A total of 81,000 samples were collected from three case studies involving SKF7205, SKF7206, and SKF7207 rolling element bearings under varying fault conditions. Feature selection using Principal Component Analysis (PCA) and correlation-based filtering was employed to reduce redundancy and enhance model performance. The classifiers were evaluated across multiple metrics including validation accuracy, test accuracy, misclassification cost, training time, and area under the the receiver operating characteristic (ROC) curve (AUC). Ensemble Bagged Trees consistently outperformed other models across all case studies, demonstrating superior classification accuracy, robustness, and low misclassification cost. Fine Tree models also demonstrated competitive performance while maintaining low computational demand, while Wide Neural Networks exhibited high predictive performance with longer training times. This work provides a practical reference for researchers and practitioners by systematically comparing ML algorithms and elucidating the trade-offs between predictive accuracy, computational efficiency, and deployment readiness in real-world fault diagnosis applications.

INDEX TERMS Bearing fault diagnosis, ML algorithm, Feature selection, Principal component analysis (PCA), Predictive maintenance, Industrial deployment, Model explainability.

I. INTRODUCTION

Modern machinery is being developed with increasingly complex architectures and is expected to operate under dynamic and often unpredictable conditions. This rising complexity makes such systems more susceptible to failures—particularly unexpected breakdowns—which in turn lead to unplanned maintenance, increased operational costs, reduced system availability, and decreased productivity. As such systems become integral to mission-critical applications, condition monitoring has emerged as a critical aspect of system reliability. Accurate and timely health monitoring not only enhances availability and operational efficiency but also reduces downtime, thereby improving overall customer satisfaction.

Rolling element bearings are fundamental components in rotating machinery. Their failure can trigger cascading malfunctions, resulting in severe system damage. Studies have shown that bearing-related defects account for approximately 45% to 55% of all failures in rotating machines [1]. Consequently, effective fault detection and classification of bearing defects have become vital for predictive maintenance strategies. A typical fault diagnosis framework comprises three stages: data acquisition, feature extraction, and fault classification. This structured pipeline enables the extraction of meaningful information from raw vibration signals and the use of intelligent classifiers to identify fault conditions [2] – [5].

Among various condition monitoring techniques, vibration analysis remains the most prevalent, particularly in sectors such as aerospace, automotive, manufacturing, and power generation [6]. Even under healthy conditions, rolling bearings exhibit vibrations due to internal dynamics such as variable compliance. Accurately differentiating these inherent signals from fault-induced anomalies is essential for reliable fault identification. Common statistical indicators such as Root Mean Square (RMS), kurtosis, crest factor, impulse factor, and shape factor are widely used for this purpose [7]. They are often employed as input features to machine learning (ML) models.

Machine learning has shown substantial promise in the automated diagnosis of mechanical faults. In supervised learning, labelled datasets are used to train algorithms such as Support Vector Machines (SVMs), Artificial Neural Networks (ANNs), and k-Nearest Neighbors (k-NN). In contrast, unsupervised learning methods identify inherent patterns or clusters in unlabelled data. Advanced techniques, including Radial Basis Function (RBF) networks [8], wavelet-transformed neural classifiers [9], and autoencoders, have further improved the accuracy and robustness of fault diagnosis models.

Several research efforts have applied ML to bearing fault classification. For instance, Kankar et al. [10] utilized SVM and ANN models to classify ball-bearing defects, while Parmar and Pandya [11] experimentally validated the performance of SVMs for cylindrical roller-bearing faults. Tools like MATLAB have also enabled integrated workflows for signal acquisition, feature extraction, and classification [12]. In a related effort, Li et al. [13] combined RBF networks with autoregressive modeling to recognize failure patterns in antifriction bearings, achieving over 90% accuracy. Fatima et al. [14] demonstrated the effectiveness of SVMs in diagnosing rotor–bearing system faults. Similarly, Samal et al. [15] proposed an AI-enhanced framework for fault diagnosis in rolling element bearings, emphasizing a comprehensive vibration-based approach. In another study, Samal et al. [16] implemented artificial neural networks for fault classification based on vibration signatures, showcasing promising results in experimental settings.

Despite such advances, a comprehensive and systematic performance evaluation of multiple ML classifiers tailored specifically for bearing fault classification remains lacking. Most existing studies focus on a limited subset of algorithms and often neglect crucial performance metrics such as accuracy, area under the receiver operating characteristic (ROC) curve (AUC), training time, and misclassification cost. While benchmarking studies are abundant in other fields—such as sentiment classification using RoBERTa and RNNs [17], [18], Parkinson's disease diagnosis using feature selection [19], [20], and COVID-19 detection via neural-network-based autoencoders [21], [22]—few efforts apply such rigorous comparison methodologies to bearing diagnostics.

Recent literature has highlighted the use of hybrid classifiers in big data applications [23], [24], vision transformers in medical image processing [25], [26], and ensemble learning for smart city infrastructure [27], [28]. Although benchmarking of ML models has been conducted in domains like urban pattern recognition [29], network intrusion detection [30], and credit card fraud detection [31], there remains a substantial gap in applying this breadth of evaluation to bearing fault classification using vibration signals.

To the best of the authors' knowledge, no prior study offers a detailed and comparative performance analysis of a wide range of ML algorithms using real-world vibration datasets from rolling element bearings.

This study aims to bridge that gap through the following key contributions:

1. **Comprehensive Benchmarking of ML Algorithms for Bearing Fault Diagnosis:** This study evaluates 33 machine learning classifiers on a large-scale, real-world vibration dataset comprising 81,000 samples collected from three types of rolling element bearings—SKF7205, SKF7206, and SKF7207. It stands as one of the most exhaustive comparative analyses in the field of bearing fault classification.
2. **Deployment-Oriented Performance Evaluation Framework:** In addition to classification accuracy, the evaluation incorporates deployment-critical factors such as misclassification cost, training time, and area under the ROC curve (AUC). This ensures a realistic and application-centric assessment of each model's industrial feasibility.
3. **Identification of Optimal Models for Practical Use:** The study identifies three top-performing models based on validation accuracy, test accuracy, and AUC. Furthermore, it highlights models that offer an optimal balance between predictive performance and computational efficiency—ideal for real-time or resource-constrained deployments.
4. **Dimensionality Reduction for Enhanced Generalization and Efficiency:** The application of Principal Component Analysis (PCA) and correlation-based feature selection reduces input redundancy and improves model generalization, thereby enabling faster and more efficient model training and inference.
5. **Actionable Guidance for Industrial Fault Monitoring Systems:** The study serves as a practical reference for engineers and practitioners aiming to deploy ML-based fault classification in rotating machinery. It offers actionable insights for selecting algorithms based on performance trade-offs, resource availability, and application-specific constraints.

The remainder of the paper is organized into three additional sections. Section 2 outlines the methodology followed in this investigation, including the experimental procedure, the acquisition of vibration data from the test bearings, the extraction of statistical features and their selection, the training of the ML algorithms, and their performance evaluation. The results obtained, along with various comparison metrics, are presented and discussed in Section 3. Finally, Section 4 concludes the investigation with final remarks.

II. METHODOLOGY

This deployment-oriented benchmarking study adopts a structured three-phase methodology: (i) vibration data acquisition under controlled experimental conditions, (ii) feature extraction based on time-domain statistical descriptors, and (iii) training, validation, and comparison of multiple machine learning (ML) models. The aim is to establish a reliable and practical evaluation framework for fault classification in anti-friction bearings.

A. EXPERIMENTATION AND DATA ACQUISITION

To ensure real-world relevance and generalizability, a comprehensive dataset was developed using a flexible experimental platform capable of simulating diverse bearing conditions. Vibration signals were acquired from multiple bearing types under varying speeds and fault scenarios. These data form the foundation for the benchmarking of machine learning algorithms.

1) EXPERIMENTAL SETUP

The experimental test rig, depicted in Fig. 1, consists of a 0.5 HP electric motor connected to a shaft via a flexible coupling, which compensates for minor misalignments. The shaft is supported by two self-aligning support bearings and includes a replaceable test bearing. An accelerometer mounted on the test bearing housing captures vibration signals, which are recorded using a NI9234 DAQ card installed on a cDAQ-9178 chassis, interfaced with a computer via NI LabVIEW software.

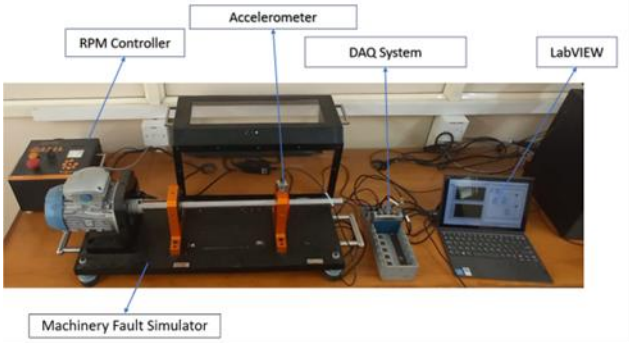


FIGURE 1. Experimental setup used for vibration signal acquisition.

2) SELECTION OF TEST BEARINGS AND FAULT INDUCTION

To assess model robustness across different bearing geometries, three rolling element bearings were selected: SKF7205, SKF7206, and SKF7207, with bore diameters of 25 mm, 30 mm, and 35 mm, respectively. For each bearing, the three health conditions, namely healthy (HB), inner race defect (IRD), and outer race defect (ORD) were investigated. Faults were introduced using Electric Discharge Machining (EDM), a precise sparking technique that simulates real-world localized defects. The process is illustrated in Fig. 2.

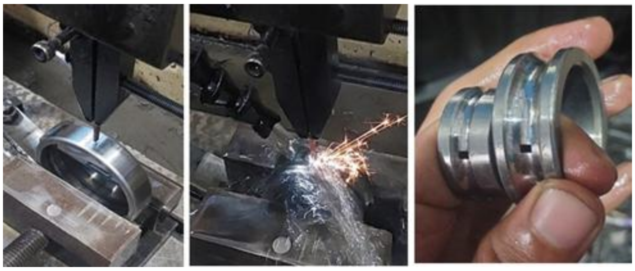


FIGURE 2. Fault induction using EDM sparking.

3) DESIGN OF EXPERIMENTS (DOE) USING TAGUCHI METHOD

Taguchi L27 orthogonal array was used to design the experiments efficiently while covering all factor combinations (Table 1). The three experimental control factors were:

- Bearing model No.: SKF7205, SKF7206, SKF7207
- Bearing condition: HB, ORD, IRD
- Rotational speed: 500 RPM, 1000 RPM, 1500 RPM

TABLE 1
L27 TAGUCHI ORTHOGONAL ARRAY USED FOR EXPERIMENTAL DESIGN

Sl. No.	Bearing Model	Bearing Condition	Shaft Speeds (RPM)
1	SKF 7205	HB	500
2	SKF 7205	HB	1000
3	SKF 7205	HB	1500
4	SKF 7205	ORD	500
5	SKF 7205	ORD	1000
6	SKF 7205	ORD	1500
7	SKF 7205	IRD	500
8	SKF 7205	IRD	1000
9	SKF 7205	IRD	1500
10	SKF 7206	HB	500
11	SKF 7206	HB	1000
12	SKF 7206	HB	1500
13	SKF 7206	ORD	500
14	SKF 7206	ORD	1000
15	SKF 7206	ORD	1500
16	SKF 7206	IRD	500
17	SKF 7206	IRD	1000
18	SKF 7206	IRD	1500
19	SKF 7207	HB	500
20	SKF 7207	HB	1000
21	SKF 7207	HB	1500
22	SKF 7207	ORD	500
23	SKF 7207	ORD	1000
24	SKF 7207	ORD	1500
25	SKF 7207	IRD	500
26	SKF 7207	IRD	1000
27	SKF 7207	IRD	1500

Each configuration was repeated five times, and vibration signals were captured for each trial. This resulted in a comprehensive dataset across three case studies, capturing both structural variability (due to different bearings) and dynamic variability (due to varying speeds and fault conditions). Such diversity ensures realistic benchmarking of the machine learning models in a deployment context.

B. EXTRACTION OF STATISTICAL FEATURE

The statistical features include basic mean, standard deviation, and root mean square (RMS) metrics. The form factor, higher-order kurtosis, and skewness statistics are also included in the feature set. It is reasonable to anticipate that each of the above features will shift when a failing fault signature encroaches on the nominal signal. From the acquired vibration data, seven statistical features, such as crest factor (CF), kurtosis (KU), impulse factor (IF), RMS, peak-to-peak (PP), skewness value (SW), and energy, were extracted using a MATLAB program.

Kurtosis: The fourth moment of a distribution normalized by the fourth power of the standard deviation is known as kurtosis. Equation (1) can be utilized to calculate this balanced measure, which evaluates the shape and tail behaviour of a dataset by combining lower and higher moments. Kurtosis is quite useful in fault diagnosis [32], [33].

$$Kurtosis\ (KU) = \frac{\sum_{i=1}^{N}(y_i - \bar{y})^4}{N * \sigma^4} \qquad (1)$$

where y_i is the instant amplitude, ȳ is the mean, and *σ,* given in (2), is the standard deviation of the data, and N is the sample length.

$$\sigma = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_i - \bar{y})^2} \qquad (2)$$

Crest Factor: The crest factor is the ratio of peak value over RMS value (3) [34].

$$Crest\ Factor\ (CF) = \frac{\max_i |y_i|}{\sqrt{\frac{1}{N}\sum_{i=1}^{N}|y_i|^2}} \qquad (3)$$

Root Mean Square (RMS): The RMS measures the overall level of a discrete signal (4).

$$RMS = \sqrt{\frac{1}{N}\sum_{i=1}^{N}|y_i|^2} \qquad (4)$$

where the number of discrete points, or the signal from each sampled point, is denoted by 'N'. These RMS values are used to evaluate a bearing's condition by comparing it to suggested standards [35].

Impulse Factor: The height of a peak is compared to the signal's mean level using the impulse factor. Peak value divided by the absolute value's mean (5).

$$Impulse\ Factor\ (IF) = \frac{\max_i |y_i|}{\frac{1}{N}\sum_{i=1}^{N}|y_i|} \qquad (5)$$

The impulse factor provides insights into the impact or shock component within a vibration signal. A high impulse factor suggests the presence of short, sharp, and intense transient events in the signal, which can be important for detecting certain types of mechanical faults or abnormalities in machinery. It is commonly used in applications where sudden impacts or shocks are of concern, such as in the analysis of bearing conditions or gear faults.

Peak–to–Peak Amplitude: Peak-to-peak amplitude is determined by measuring the distance from the highest positive peak to the lowest negative peak.

$$peak2Peak\ Amplitude\ (PP) = y_{max} - y_{min} \qquad (6)$$

Skewness: A signal distribution's asymmetry is skewness. Flaws have the potential to disrupt distribution symmetry, which raises the degree of skewness. (7).

$$Skewness\ (SW) = \frac{\sum_{i=1}^{N}(y_i - \bar{y})^3}{N * \sigma^3} \qquad (7)$$

Energy: Energy is calculated using (8) as the squared mean of the square root of the absolute values of the discrete signal data [36].

$$Enenrgy = \left(\frac{1}{N}\sum_{i=1}^{N}|y_i|^{1/2}\right)^2 \qquad (8)$$

C. FEATURE SELECTION

To statistically justify the selection of the seven time-domain features—RMS, Kurtosis, Crest Factor, Impulse Factor, Peak-to-Peak, Skewness, and Energy—Principal Component Analysis (PCA) and correlation analysis were performed on the standardized dataset.

D. TRAINING THE MACHINE LEARNING ALGORITHMS

All the classification learners, or machine learning algorithms for classification available in the MATLAB environment, were trained and tested using vibration data to evaluate their performance in bearing fault classification. These algorithms are grouped into ten broad categories: decision trees, discriminant analysis, logistic regression classifiers, Naive Bayes classifiers, support vector machines (SVM), efficiently trained linear classifiers, nearest neighbor's classifiers, kernel approximation classifiers, ensemble classifiers, and neural network classifiers.

Variants Within the Ten Categories of ML Algorithms:

1. **Decision Trees:** This classifier is known for being easy to interpret, fast in fitting and prediction, and low on memory usage. It segments the population into branches, forming an inverted tree structure with root, internal, and

leaf nodes. The Fine, Medium, and Coarse Trees are the three variants [37].

2. **Discriminant Analysis:** This classification scheme assumes that unique Gaussian distributions for each class are used in data generation. This classification method assumes that different classes generate data based on distinct Gaussian distributions. The two variants are Linear and quadratic discriminants.
3. **Logistic Regression (LR):** A machine learning classifier that estimates the likelihood of a binary response based on input features by determining the optimal fit parameters.
4. **Naive Bayes (NB):** A highly efficient classification technique that applies the Bayes theorem under the assumption of strong (naive) independence among features. The two variants are Gaussian NB and Kernel NB [38].
5. **Support Vector Machines (SVM):** An effective classifier that finds an optimal hyperplane in the feature space, maximizing the margin between classes. This approach handles high-dimensional data well and is less prone to overfitting with properly tuned C parameters. The six variants available in MATLAB are Linear, Quadratic, Cubic, Fine Gaussian, Medium Gaussian, and Coarse Gaussian SVM [39] – [42]
6. **Efficiently Trained Linear Classifiers:** These classifiers reduce training computation time, albeit with some accuracy trade-off. Available models include logistic regression and support vector machines (SVM)
7. **K-Nearest Neighbors (KNN):** A simple yet powerful algorithm for classification and regression tasks. It predicts the class of a data point by identifying the ’k’ nearest neighbors and taking a majority vote. The six variants are Fine, Medium, Coarse, Cosine, Cubic, and Weighted KNN [43], [44].
8. **Kernel Approximation Classifiers:** These classifiers perform nonlinear classification of large datasets. They tend to train and predict faster than SVM classifiers with Gaussian kernels for large in-memory data.
9. **Ensemble Classifiers:** These combine the results from multiple weak learners to form a high-quality ensemble model. The ensemble classifiers have five variants, namely, boosted trees, bagged trees, subspace discriminant, subspace KNN, and RUS-boost trees, with bagged trees using Breiman’s ’random forest’ algorithm.
10. **Neural Network (NN) Classifiers:** These models are known for good predictive accuracy and are suited for multiclass classification. Model flexibility increases with the size and number of fully connected layers in the network. The five variants are Narrow NN, Medium NN, Wide NN, Bilayered NN, and Trilayered NN. Each is a feedforward, fully connected neural network where each subsequent layer is connected to the previous one, with each layer multiplying the input by a weight matrix and adding a bias vector estimated [45], [46].

All the aforementioned ML algorithms have been trained and tested with the comprehensive dataset generated by extracting the seven statistical features from the experimental bearing vibration data. Seventy percent of the dataset has been used for training algorithms, with the remaining thirty percent reserved for testing their performance in bearing fault classification.

E. HYPERPARAMETER SETTINGS AND CROSS-VALIDATION STRATEGY

All 33 machine learning algorithms, spanning ten classifier categories as described earlier, are trained using MATLAB’s Classification Learner App with default hyperparameter configurations. To ensure reliable performance evaluation and minimize overfitting, a 5-fold cross-validation strategy is uniformly applied across all models during the training phase. Detailed cross-validation statistics and variability measures are evaluated for the three top-performing models.

F. PERFORMANCE EVALUATION OF THE MACHINE LEARNING (ML) ALGORITHMS

Seven time-domain statistical features—root mean square (RMS), kurtosis, crest factor, impulse factor, peak-to-peak value, skewness, and energy—were extracted from the vibration signals and used as predictor variables, with the three bearing health conditions (healthy bearing (HB), outer race defect (ORD), and inner race defect (IRD)) serving as target classes. Classifier performance was evaluated using training and testing results, confusion matrix-derived metrics, and receiver operating characteristic (ROC) curves.

The confusion matrix serves as a diagnostic tool that helps identify areas where the classifier's performance is suboptimal due to misclassification between response classes. The Positive Predictive Value (PPV) quantifies the proportion of correctly predicted positive instances among all predicted positives, thereby indicating the classifier's precision. Conversely, the False Discovery Rate (FDR), defined as one minus the PPV, reflects the probability that a predicted positive instance is, in fact, a false positive. The True Positive Rate (TPR), also referred to as sensitivity or recall, denotes the proportion of correctly classified instances within each actual class, whereas the False Negative Rate (FNR) measures the proportion of misclassified instances within each true class. Finally, the average of the TPR values across all classes is computed to determine the overall accuracy of the classifier.

Misclassification costs were calculated using a unit cost approach, with a cost of one for each incorrect classification and zero for correct predictions. ROC curves, which plot TPR against false positive rate (FPR), were used to derive the area under the curve (AUC), a widely accepted measure of classifier discriminative ability.

G. STATISTICAL SIGNIFICANCE TEST USING WILCOXON SIGNED-RANK TEST

To assess whether the differences in model performance between the validation and test phases were statistically significant, we conducted a non-parametric Wilcoxon signed-rank test across all 33 machine learning algorithms.

H. MODEL EXPLAINABILITY APPROACH

To enhance the interpretability of the machine learning models and address the practical relevance of fault classification, a comprehensive model explainability strategy was employed. This included feature importance analysis, partial dependence plots (PDPs), sensitivity analysis via feature perturbation, and confusion matrices applied to the top performing models.

1) FEATURE IMPORTANCE ANALYSIS

Feature importance methods were used to rank input vibration features based on their relative contribution to model predictions. Techniques such as Minimum Redundancy Maximum Relevance (mRMR) and ReliefF algorithms were utilized to identify the most discriminative features.

mRMR (Minimum Redundancy Maximum Relevance) selects features that are maximally relevant to the target and minimally redundant to other features. It emphasizes information gain and independence.

ReliefF evaluates feature importance based on the ability to distinguish between classes, considering local neighborhoods. It detects feature interactions and nonlinear dependencies.

2) PARTIAL DEPENDENCE PLOTS (PDPS) FOR ENSEMBLE BAGGED TREE MODEL

Partial Dependence Plots (PDPs) estimate the marginal effect of a feature on the model's output by averaging predictions across the distribution of all other features. While this approach does not hold other features constant in the strict sense, it approximates the isolated influence of the selected feature. In this study, PDPs were employed to visualize how changes in individual features affect the predicted probabilities of each fault class, thereby enhancing interpretability and revealing insights into model sensitivity and decision behaviour.

3) SENSITIVITY ANALYSIS VIA FEATURE PERTURBATION FOR WIDE NEURAL NETWORK MODEL

Model explainability for neural network models, the sensitivity analysis approach was adopted to assess feature influence on model predictions. To evaluate how sensitive the neural network's output is to individual input features, each feature was independently perturbed by a fixed percentage (e.g., ±10%), and the change in model output was recorded. This method provides a practical insight into the relative importance of input features by highlighting which ones cause the most significant changes in the output when slightly altered.

4) CONFUSION MATRICES

Confusion matrices provided class-wise evaluation by visualizing the distribution of predicted versus actual labels. This enabled a granular assessment of model strengths and limitations in distinguishing specific fault categories, supporting a more transparent evaluation of diagnostic performance.

III. RESULTS AND DISCUSSION

As detailed in the section 'Experimentation and Data Acquisition,' the experiments were conducted following the Design of Experiments (DoE) outlined in Table 1. For each of the twenty-seven combinations of bearing diameters, bearing conditions, and shaft speeds, experiments were repeated five times, resulting in a total of 135 experiments.

From the vibration data collected in each experiment, 600 sets of seven statistical features were extracted, resulting in 3,000 sets of data for each of the 27 cases outlined in Table 1. This process generated a total of 81,000 data sets.

A. FEATURE SELECTION USING PRINCIPAL COMPONENT ANALYSIS (PCA) AND CORRELATION ANALYSIS

To statistically justify the selection of the seven time-domain features—RMS, Kurtosis, Crest Factor, Impulse Factor, Peak-to-Peak, Skewness, and Energy—Principal Component Analysis (PCA) and correlation analysis were performed on the standardized dataset.

The explained variance ratio (%) for each principal component is summarized in Table 2. The scree plot is shown in Fig. 3 and the component loading matrix is presented in Table 3.

TABLE 2
EXPLAINED VARIANCE RATIO (%) FOR EACH PRINCIPAL COMPONENTS

Principal Components	PC1	PC2	PC3	PC4	PC5	PC6	PC7
	68.58	**20.73**	**7.41**	1.72	0.73	0.46	0.37

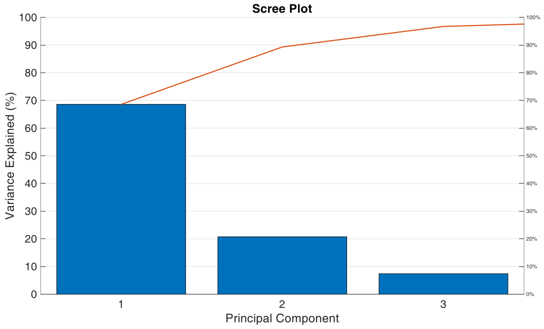


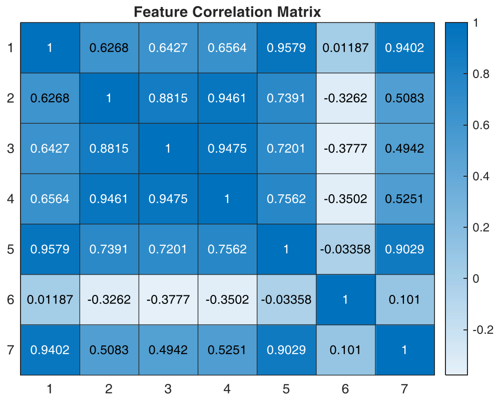
FIGURE 3. Scree plot showing the percentage of variance explained by each principal component. The first three components capture more than 96% of the total variance.

TABLE 3
PRINCIPAL COMPONENT LOADING SCORES OF THE STANDARDIZED FEATURES. SIGNIFICANT CONTRIBUTIONS (>0.3) ARE BOLDED TO HIGHLIGHT FEATURE IMPORTANCE ACROSS PCS.

Feature	PC1	PC2	PC3	PC4	PC5	PC6	PC7
RMS	**0.41**	**0.32**	-0.26	0.16	**-0.39**	**-0.30**	**0.63**
Kurtosis	**0.41**	-0.24	**0.34**	**-0.68**	0.02	**-0.45**	-0.03
Crest Factor	**0.41**	-0.27	0.25	**0.70**	0.13	**-0.33**	-0.28
Impulse Factor	**0.42**	-0.25	**0.30**	0.02	0.15	**0.69**	**0.41**
Peak to Peak	**0.43**	0.23	-0.12	-0.09	**-0.53**	**0.34**	**-0.59**
Skewness	-0.11	**0.68**	**0.72**	0.06	0.02	-0.01	0.01
Energy	**0.36**	**0.43**	**-0.36**	-0.09	**0.73**	0.01	-0.12

The analysis revealed that the first three principal components captured 96.72% of the total variance, with PC1 alone accounting for 68.59%. The corresponding loading scores for PC1 show that RMS, Kurtosis, Crest Factor, Impulse Factor, Peak-to-Peak, and Energy contribute significantly and nearly equally to the primary source of variance. Although Skewness contributes minimally to PC1, it is dominant in PC2 and PC3, with loading values of 0.683 and 0.720, respectively, indicating its role in capturing orthogonal variation relevant to fault classification.

In addition to PCA, a Pearson correlation analysis was performed to assess inter-feature redundancy. The correlation matrix is presented in Fig. 4.



	1	2	3	4	5	6	7
1	1	0.6268	0.6427	0.6564	0.9579	0.01187	0.9402
2	0.6268	1	0.8815	0.9461	0.7391	-0.3262	0.5083
3	0.6427	0.8815	1	0.9475	0.7201	-0.3777	0.4942
4	0.6564	0.9461	0.9475	1	0.7562	-0.3502	0.5251
5	0.9579	0.7391	0.7201	0.7562	1	-0.03358	0.9029
6	0.01187	-0.3262	-0.3777	-0.3502	-0.03358	1	0.101
7	0.9402	0.5083	0.4942	0.5251	0.9029	0.101	1

FIGURE 4. Pearson correlation matrix of the seven selected time-domain features.

Strong positive correlations were observed between RMS, Peak-to-Peak, Energy, Crest Factor, and Impulse Factor (e.g., RMS–Peak-to-Peak: r = 0.9579; Crest Factor–Impulse Factor: r = 0.9475), reflecting shared amplitude-related signal characteristics. Conversely, Skewness displayed weak or negative correlations with all other features (e.g., Skewness–RMS: r = 0.01187; Skewness–Crest Factor: r = -0.3777), suggesting that it offers non-redundant, complementary information that enhances the discriminatory power of the feature set.

Overall, the results from PCA and correlation analysis jointly validate the use of all seven statistical features, supporting their relevance and effectiveness for robust vibration-based fault classification.

B. PERFORMANCE EVALUATION OF MACHINE LEARNING (ML) ALGORITHMS

From a practical application perspective, the 81,000 data sets were divided into three groups based on different bearing numbers, reflecting the typical condition monitoring process for a specific bearing in operation. Specifically, for SKF7205, SKF7206, and SKF7207 bearings, 27,000 data sets each were allocated for training and testing the algorithms. This approach allows for three distinct case studies to evaluate the performance of these ML algorithms.

A total of 33 machine learning models were implemented, and for each, the validation accuracy, test accuracy, total cost, AUC, and training time were analyzed. The performance of these ML algorithms across the three case studies is discussed as follows.

1) CASE STUDY 1: SKF7205 BEARING FAULT CLASSIFICATION

All 33 classifiers were trained using 70% of the data (18,900 sets) and tested with the remaining 30% (8,100 sets). The performance metrics for all 33 models are summarized in Table 4.

This can be observed from Table 4 that the Ensemble Bagged Trees model achieved the highest validation accuracy of 96.14%, followed by the Fine Tree at 95.70% and the Wide Neural Network at 94.31%. Other top-performing models included Ensemble Boosted Trees, Medium Neural Network, Ensemble Subspace KNN, Fine Gaussian SVM, Bi-layered Neural Network, Tri-layered Neural Network, and Narrow Neural Network—all with validation accuracies exceeding 91%.

During the testing phase, the top three models maintained their ranks. However, the Medium Neural Network and Fine Gaussian SVM improved their positions, and the Weighted KNN model entered the top 10. Conversely, the Narrow Neural Network dropped to 13th position.

Misclassification cost rankings closely mirrored accuracy-based rankings, confirming that lower cost was associated with higher accuracy. Regarding AUC, Ensemble Bagged Trees led during validation, followed by the Wide Neural Network and Ensemble Boosted Trees. In the testing phase, the Wide Neural Network showed superior class discrimination with the highest AUC.

In terms of computational efficiency, the Fine Tree required the least training time (8.76 s), whereas the Cubic SVM and Quadratic SVM were the most computationally expensive, requiring 1690.1 s and 1460.3 s, respectively. Notably, neural network models had higher training durations despite their strong performance.

2) CASE STUDY 2: SKF7205 BEARING FAULT CLASSIFICATION

All 33 classifiers were trained using 70% of the data (18,900 sets) and tested with the remaining 30% (8,100 sets). The performance metrics for all 33 models are summarized in Table 5.

As observed from Table 5, Ensemble Bagged Trees once again outperformed other models with a validation accuracy of 98.03%, followed closely by the Wide Neural Network (97.81%) and Fine Tree (97.79%). Other competitive models included the Medium Neural Network, Bi-layered Neural Network, and Cubic SVM, each achieving validation accuracies above 97.5%.

Testing results closely reflected validation trends. The Wide Neural Network achieved the highest test accuracy (98.07%), followed by Ensemble Bagged Trees (98.06%) and Fine Tree (97.77%).

In terms of validation cost, Ensemble Bagged Trees incurred the lowest cost (372), reaffirming its classification efficiency. Although neural networks demonstrated high accuracy, they tended to incur higher misclassification costs due to occasional prediction inconsistencies. During testing, the Wide Neural Network had the lowest cost (156), slightly better than Ensemble Bagged Trees (157) and Fine Tree (174).

The AUC values during validation were highest for Ensemble Bagged Trees, Wide Neural Network, and Cubic SVM, indicating robust classification boundaries. The Wide Neural Network further improved its AUC during testing, confirming its strong generalization performance.

From the computational perspective, the Fine Tree had the shortest training time (7.8 s), while the Wide Neural Network and Cubic SVM required significantly longer durations (698.42 s and 1623.1 s, respectively).

3) CASE STUDY 3: KSF7207 BEARING FAULT CLASSIFICATION

Similar to the previous case studies, all 33 classifiers were trained using 70% of the data (18,900 sets) and tested with the remaining 30% (8,100 sets). The comprehensive performance metrics for all 33 models are summarized in Table 6.

TABLE 4
PERFORMANCE METRICS FOR FAULT CLASSIFICATION OF SKF7205 BEARING

Model No.	Model	Validation			Testing			Training Time (s)
		Accuracy (%)	Total Cost	ROC (AUC)	Accuracy (%)	Total Cost	ROC (AUC)	
1	Fine Tree	95.7	813	0.990	95.52	363	0.988	8.7577
2	Medium Tree	91.46	1614	0.972	91.09	722	0.970	13.279
3	Coarse Tree	77.34	4282	0.865	77.01	1862	0.865	11.887
4	Linear Discriminant	72.91	5120	0.833	72.9	2195	0.835	11.451
5	Quadratic Discriminant	79.92	3795	0.916	80.07	1614	0.921	10.848
6	Efficient Logistic Regression	77.33	4284	0.887	78.12	1772	0.892	10.04
7	Efficient Linear SVM	79.13	3945	0.884	79.72	1643	0.892	9.5771
8	Gaussian Naive Bayes	73.51	5006	0.840	73.6	2138	0.845	8.9378
9	Kernel Naive Bayes	77.72	4210	0.884	77.83	1796	0.890	134.16
10	Linear SVM	79.53	3868	0.890	80.16	1607	0.895	304.67
11	Quadratic SVM	90.37	1820	0.974	90.33	783	0.976	1460.3
12	Cubic SVM	78.79	4008	0.937	72.07	2262	0.938	1690.1
13	Fine Gaussian SVM	92.8	1360	0.982	93.52	525	0.985	169.51
14	Medium Gaussian SVM	89.13	2054	0.968	89.57	845	0.973	196.1
15	Coarse Gaussian SVM	79.35	3903	0.920	80.48	1581	0.928	238.07
16	Fine KNN	91.02	1697	0.933	91.63	678	0.937	239.61
17	Medium KNN	91.14	1675	0.982	91.85	660	0.985	241.39
18	Coarse KNN	87.99	2270	0.979	88.62	922	0.983	245.36
19	Cosine KNN	88.41	2191	0.973	89.26	870	0.976	251.89
20	Cubic KNN	90.91	1718	0.982	91.73	670	0.984	258.42
21	Weighted KNN	91.76	1557	0.983	92.58	601	0.986	259.85
22	Ensemble Boosted Trees	93.93	1148	0.991	93.32	541	0.990	276.45
23	Ensemble Bagged Trees	96.14	730	0.994	95.77	343	0.992	392.18
24	Ensemble Subspace Discriminant	74.96	4733	0.832	74.59	2058	0.833	307.59
25	Ensemble Subspace KNN	93.14	1297	0.989	93.26	546	0.988	316.68
26	Ensemble RUS Boosted Trees	91.46	1614	0.972	91.09	722	0.970	312.03
27	Narrow Neural Network	91.9	1531	0.985	91.65	676	0.982	373.59
28	Medium Neural Network	93.56	1217	0.991	93.85	498	0.992	425.91
29	Wide Neural Network	94.31	1076	0.992	95.04	402	0.993	678.52
30	Bilayered Neural Network	92.78	1364	0.987	93.38	536	0.989	526.03
31	Trilayered Neural Network	92.51	1415	0.987	91.98	650	0.985	653.06
32	SVM Kernel	82.58	3292	0.942	85.95	1138	0.960	677.37
33	Logistic Regression Kernel	82.46	3316	0.946	80.43	1585	0.931	684.91

TABLE 5
PERFORMANCE METRICS FOR FAULT CLASSIFICATION OF SKF7206 BEARING

Model No.	Model	Validation			Testing			Training Time (s)
		Accuracy (%)	Total Cost	ROC (AUC)	Accuracy (%)	Total Cost	ROC (AUC)	
1	Fine Tree	97.63	447	0.994	97.73	184	0.994	5.541
2	Medium Tree	93.80	1172	0.986	95.26	384	0.988	4.427
3	Coarse Tree	81.63	3472	0.912	82.10	1450	0.914	2.6684
4	Linear Discriminant	79.25	3921	0.926	79.07	1695	0.923	2.8365
5	Quadratic Discriminant	77.50	4252	0.926	77.74	1803	0.928	2.8141
6	Efficient Logistic Regression	72.44	5209	0.890	70.90	2357	0.870	3.8799
7	Efficient Linear SVM	74.02	4911	0.856	73.64	2135	0.885	2.2372
8	Gaussian Naive Bayes	63.71	6859	0.848	64.23	2897	0.845	2.3662
9	Kernel Naive Bayes	78.41	4080	0.962	78.89	1710	0.964	519.04
10	Linear SVM	86.48	2556	0.971	86.74	1074	0.972	34.757
11	Quadratic SVM	94.92	960	0.995	95.98	326	0.997	591.16
12	Cubic SVM	97.78	420	0.999	97.59	195	0.999	6763.5
13	Fine Gaussian SVM	96.63	636	0.997	96.57	278	0.997	55.103
14	Medium Gaussian SVM	94.16	1103	0.993	94.37	456	0.994	72.006
15	Coarse Gaussian SVM	85.94	2657	0.969	86.86	1064	0.973	95.251
16	Fine KNN	94.74	994	0.961	95.10	397	0.963	97.52
17	Medium KNN	94.34	1070	0.993	93.96	489	0.993	99.27
18	Coarse KNN	90.67	1764	0.985	91.20	713	0.986	103.17
19	Cosine KNN	89.65	1956	0.980	90.12	800	0.983	109.57
20	Cubic KNN	93.07	1310	0.994	93.04	564	0.990	115.01
21	Weighted KNN	95.33	882	0.995	95.52	363	0.995	117.3
22	Ensemble Boosted Trees	97.24	521	0.998	97.20	227	0.999	510.27
23	Ensemble Bagged Trees	97.94	390	0.998	98.02	160	0.999	527.94
24	Ensemble Subspace Discriminant	74.17	4882	0.911	75.16	2012	0.944	524.58
25	Ensemble Subspace KNN	95.66	820	0.995	95.68	350	0.995	539.29
26	Ensemble RUS Boosted Trees	93.80	1171	0.990	95.26	384	0.992	542.03
27	Narrow Neural Network	97.20	530	0.998	97.41	210	0.998	676.82
28	Medium Neural Network	97.57	460	0.999	98.00	162	0.999	1533.7
29	Wide Neural Network	97.69	436	0.999	97.86	173	0.999	6665
30	Bi-layered Neural Network	97.49	475	0.999	97.78	180	0.998	1675.9
31	Tri-layered Neural Network	97.57	460	0.999	97.73	184	0.999	6629.6
32	SVM Kernel	94.51	1037	0.992	91.40	697	0.992	6591.7
33	Logistic Regression Kernel	92.31	1453	0.986	91.15	717	0.983	6602.3

TABLE 6
PERFORMANCE METRICS FOR FAULT CLASSIFICATION OF SKF7207 BEARING

Sl. No.	Model Type	Validation			Testing			Training Time (s)
		Accuracy (%)	Total Cost	ROC (AUC)	Accuracy (%)	Total Cost	ROC (AUC)	
1	Fine Tree	99.09	172	0.997	99.22	63	0.995	5.5888
2	Medium Tree	99.07	176	0.999	98.96	84	0.999	4.0245
3	Coarse Tree	86.85	2486	0.956	86.93	1059	0.959	1.7013
4	Linear Discriminant	76.75	4395	0.845	77.14	1852	0.852	2.4536
5	Quadratic Discriminant	86.25	2599	0.934	86.80	1069	0.936	2.6895
6	Efficient Logistic Regression	59.43	7668	0.709	59.93	3246	0.713	2.6283
7	Efficient Linear SVM	78.74	4018	0.860	80.91	1546	0.913	3.2639
8	Gaussian Naive Bayes	87.83	2301	0.865	88.10	964	0.869	3.3097
9	Kernel Naive Bayes	91.25	1653	0.985	92.04	645	0.988	109.85
10	Linear SVM	83.09	3196	0.908	83.83	1310	0.915	1217.8
11	Quadratic SVM	97.54	464	0.998	97.75	182	0.998	1475.9
12	Cubic SVM	99.06	178	1.000	98.84	94	0.999	1697.2
13	Fine Gaussian SVM	98.29	323	0.999	98.42	128	1.000	1087
14	Medium Gaussian SVM	96.33	694	0.998	97.12	233	0.998	1161.1
15	Coarse Gaussian SVM	85.04	2828	0.919	85.96	1137	0.929	1163.9
16	Fine KNN	96.56	650	0.972	97.40	211	0.978	1163.4
17	Medium KNN	95.47	856	0.996	95.90	332	0.996	1162.6
18	Coarse KNN	92.70	1379	0.992	93.32	541	0.994	1161.8
19	Cosine KNN	88.43	2187	0.971	89.21	874	0.974	1161.1
20	Cubic KNN	94.63	1014	0.994	95.11	396	0.995	1160.4
21	Weighted KNN	96.76	612	0.998	97.62	193	0.998	1159.7
22	Ensemble Boosted Trees	99.15	161	1.000	99.12	71	1.000	1162.7
23	Ensemble Bagged Trees	99.38	117	1.000	99.38	50	1.000	1176
24	Ensemble Subspace Discriminant	77.59	4236	0.808	77.86	1793	0.814	1184.1
25	Ensemble Subspace KNN	98.85	217	1.000	98.99	82	1.000	1196.8
26	Ensemble RUS Boosted Trees	99.08	173	1.000	99.01	80	1.000	1203.5
27	Narrow Neural Network	98.92	205	1.000	98.88	91	1.000	1284.5
28	Medium Neural Network	99.01	187	1.000	98.93	87	1.000	1329.3
29	Wide Neural Network	99.01	188	0.999	99.09	74	1.000	1577.4
30	Bi-layered Neural Network	98.93	202	1.000	98.95	85	1.000	1443
31	Tri-layered Neural Network	98.98	192	1.000	99.02	79	1.000	1585.3
32	SVM Kernel	89.76	1935	0.970	90.17	796	0.967	1496.6
33	Logistic Regression Kernel	88.52	2169	0.966	88.21	955	0.968	1507.5

As observed from Table 6, Ensemble Bagged Trees achieved the highest validation and test accuracies of 99.38%, with the lowest validation cost (117) and test cost (50), and a perfect AUC of 1.000, making it the most robust model overall. The Fine Tree classifier followed closely with validation/test accuracies of 99.09% and 99.22%, and slightly higher costs of 172 (validation) and 63 (test). Its AUC values of 0.997 (validation) and 0.995 (test) indicate excellent classification boundaries. Importantly, the Fine Tree required only 5.59 seconds for training, offering significant efficiency compared to other high-accuracy models, thus making it highly suitable for real-time or resource-constrained fault diagnosis applications.

The Wide Neural Network, while slightly lower in accuracy (99.01% validation, 99.09% test) and with higher training time (1577.4 seconds), exhibited excellent generalization with an AUC of 1.000 in testing and 0.999 in validation. The Trilayered Neural Network (validation/test accuracies of 98.98%/99.02%) and Bilayered Neural Network (98.93%/98.95%) also delivered strong performance but required significantly longer training times (1585.3 and 1443 seconds, respectively), which may limit their use in time-sensitive environments.

Other ensemble-based models like Boosted Trees and RUS Boosted Trees offered near-optimal accuracies (≥99.08%) with full AUC scores, but their higher training times (1162.7–1203.5 seconds) and moderate misclassification costs made them less efficient than Bagged

Trees and Fine Tree. Models like Medium Tree and Medium Neural Network provided competitive results with validation/test accuracies around 99.07%–99.01%, but the Medium Tree stood out with a very low training time of just 4.02 seconds, the fastest among all classifiers.

These results confirm that Ensemble Bagged Trees remained the most balanced classifier, excelling across all key metrics. However, in scenarios where training time is critical, Fine Tree and Medium Tree provide strong alternatives with comparable accuracy and far superior computational efficiency.

C. STATISTICAL SIGNIFICANCE TEST USING WILCOXON SIGNED-RANK TEST

To assess whether the differences in model performance between the validation and test phases were statistically significant, we conducted a non-parametric Wilcoxon signed-rank test across all 33 machine learning algorithms. This test compared the validation and test accuracies obtained for each model.

The test yielded a p-value of 0.06564, a signed-rank statistic of 177.500, and a hypothesis test result (h) = 0, indicating that there is no statistically significant difference at the 5% significance level. These results suggest that the variations in performance between the validation and test datasets are not statistically significant and therefore, support the robustness and generalizability of the models evaluated in this comparative study.

D. Comparative Analysis of Top Performing Models

A comparative analysis of the top three classifiers—Ensemble Bagged Trees, Fine Tree, and Wide Neural Network—was conducted across three bearing case studies (SKF7205, SKF7206, and SKF7207). The variation in key performance metrics across all three case studies for these top models is illustrated in Fig. 5 through Fig. 11. Additionally, a heatmap summarizing four critical metrics—validation accuracy, test accuracy, area under the validation ROC curve, and area under the test ROC curve—is presented in Fig. 12.

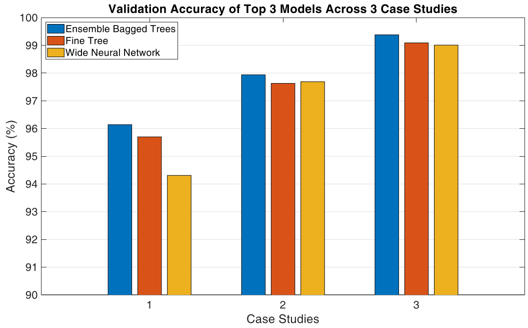


FIGURE 5. Validation accuracy comparison of top 3 models across 3 case studies.

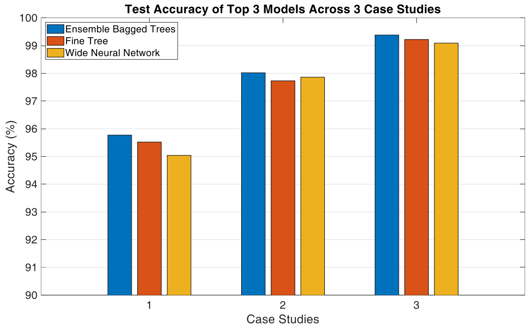


FIGURE 6. Test accuracy comparison of top 3 models across 3 case studies.

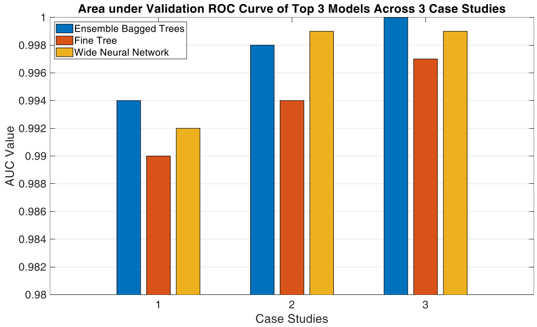


FIGURE 7. Area under validation ROC curve of top 3 models across 3 case studies.

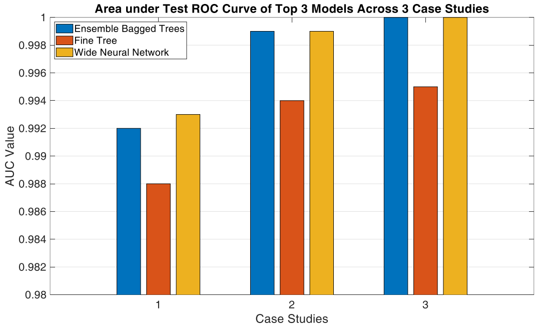


FIGURE 8. Area under test ROC curve of top 3 models across 3 case studies.

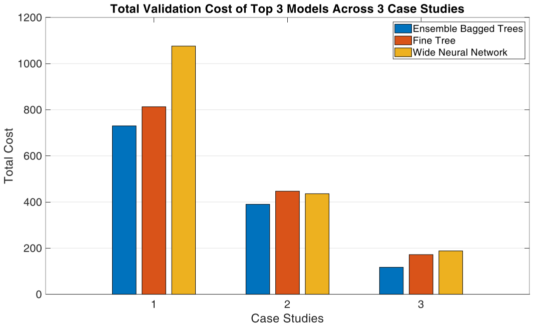


FIGURE 9. Total validation cost of top 3 models across 3 case studies.

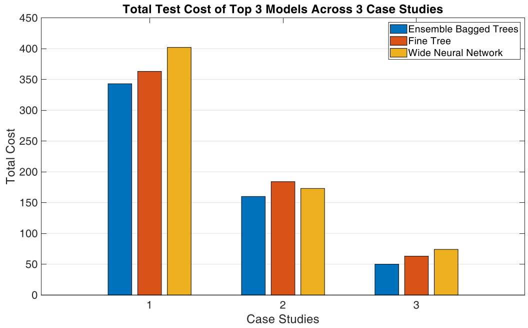


FIGURE 10. Total test cost of top 3 models across 3 case studies.

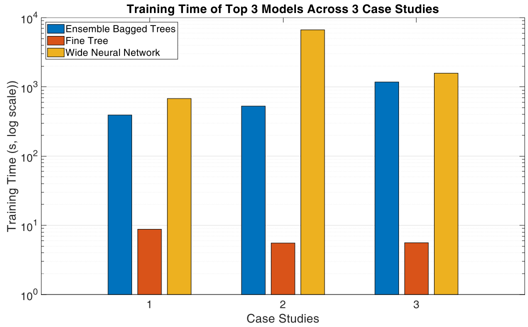


FIGURE 11. Training time of top 3 models across 3 case studies.

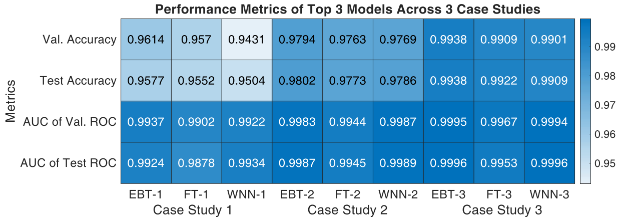


FIGURE 12. Performance metrics (validation accuracy, test accuracy, area under validation ROC curve, and area under test ROC curve) of top 3 models across 3 case studies.

This analysis revealed consistent trends in classification performance, discriminative ability (AUC), and computational efficiency among the top-performing models:

Consistent Top Performer: Ensemble Bagged Trees consistently delivered the highest overall performance, achieving validation accuracies of 96.14%, 97.94%, and 99.38% in Case Studies 1, 2, and 3, respectively, along with near-perfect AUC values (≥ 0.992) across all cases. This model also exhibited strong generalization to unseen data, with test accuracies of 95.77%, 98.02%, and 99.38%. While training times were moderate to high—particularly in Case Study 3 (1176 seconds)—the trade-off between accuracy and computational cost remained highly favourable.

Least Training Time Without Compromising Accuracy: The Fine Tree model, although slightly trailing in accuracy (ranging from 95.70% to 99.09%), demonstrated exceptionally low training times (all under 9 seconds), making it highly suitable for real-time or resource-constrained applications. Its AUC values, though marginally lower than those of ensemble models, remained above 0.988, indicating good discriminative capability.

Higher Accuracy with Increased Training Time: The Wide Neural Network achieved competitive accuracy and AUC (up to 99.01% and 1.000, respectively), especially in Case Study 3. However, this performance came at the cost of significantly higher training times, reaching over 6600 seconds in Case Study 2. This makes it less suitable for time-sensitive applications, despite its high classification precision.

Model Selection Guideline: Overall, Ensemble Bagged Trees provided the most balanced trade-off among accuracy, AUC, and training time, making it the preferred choice for practical implementations. The Fine Tree model offers a lightweight yet reasonably accurate alternative, ideal for applications where computational efficiency is critical. In contrast, Wide Neural Networks, while highly accurate, may be better suited for offline or batch-processing environments due to their substantial training overhead.

E. STATISTICAL TESTING USING K-FOLD CROSS-VALIDATION

All 33 machine learning models in this study were trained using 5-fold cross-validation. To ensure robust performance estimation, detailed cross-validation statistics and variability measures are evaluated and presented here for the three top-performing models: Ensemble Bagged Tree, Fine Tree, and Wide Neural Network. Table 7 summarizes the validation accuracy, including, per-fold accuracies, their mean, and standard deviation for these models.

TABLE 7
DETAILS OF 5-FOLD CROSS-VALIDATION OF THREE TOP PERFORMING MODELS

Model	Per-fold Accuracies (%)	Mean Accuracy (%)	Standard Deviation (%)
Ensemble Bagged Tree	96.11, 96.33, 95.72, 96.22, 96.24	96.13	0.24
Fine Tree	95.56, 95.87, 95.46, 95.50, 95.80	95.64	0.18
Wide Neural Network	95.28, 94.44, 94.85, 94.52, 95.19	94.86	0.38

The 5-fold cross-validation results demonstrate that all three top-performing models achieve high and consistent classification accuracy. The Ensemble Bagged Tree model achieved the highest mean accuracy of 96.13% with a low standard deviation of 0.24%, indicating strong and stable performance across different data splits. The Fine Tree model showed similarly reliable results with a mean accuracy of 95.64% and the lowest variability at 0.18%, reflecting excellent consistency. The Wide Neural Network, while slightly lower in mean accuracy at 94.86%, maintained acceptable stability with a standard deviation of 0.38%. These low standard deviations across models confirm minimal sensitivity to data partitioning, supporting the generalizability of the results. Overall, the data validate the robustness of these models for the fault classification task.

F. MODEL EXPLAINABILITY RESULTS

To support practical relevance and ensure transparency in model behavior, explainability techniques such as feature importance analysis, partial dependence plots (PDPs), sensitivity analysis via feature perturbation, and confusion matrices applied to the top-performing models namely, the Ensemble Bagged Trees model, Wide Neural Network model and Fine Tree model.

1) FEATURE IMPORTANCE

The importance score of the seven features determined by using the Minimum Redundancy Maximum Relevance (mRMR) algorithm is presented in Table 8 and Figure 13. Table 9 and Figure 14 represent the same determined by ReliefF algorithm.

TABLE 8
FEATURE IMPORTANCE SCORES SORTED BY USING MRMR ALGORITHM

Sl. No.	Features	Importance Scores
1	RMS	0.4177
2	Kurtosis	0.5179
3	Crest Factor	0.2180
4	Impulse Factor	0.3252
5	Peak to Peak	0.2830
6	Skewness	0.4162
7	Energy	0.1660

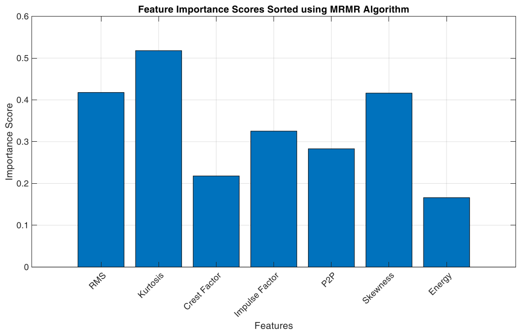


FIGURE 13. **Feature Importance Scores Sorted using Minimum Redundancy Maximum Relevance (mRMR) Algorithm.**

TABLE 9
FEATURE IMPORTANCE SCORES SORTED BY USING RELIEFF ALGORITHM

Sl. No.	Features	Importance Scores
1	RMS	0.0774
2	Kurtosis	0.0958
3	Crest Factor	0.0221
4	Impulse Factor	0.0414
5	Peak to Peak	0.0734
6	Skewness	0.0191
7	Energy	0.0708

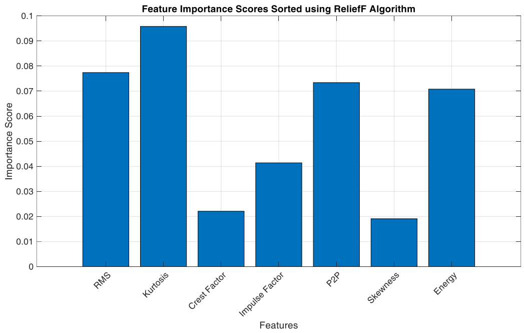


FIGURE 14. **Feature Importance Scores Sorted using RELIEFF Algorithm.**

From the results, it is evident that Kurtosis and RMS consistently emerge as the most influential features in bearing fault classification from both the mRMR and ReliefF algorithms. According to the mRMR algorithm, next to these two features, the skewness has a higher score and the remaining four features contribute almost equally, indicating a relatively uniform distribution of discriminative power. In contrast, like mRMR, ReliefF assigns higher importance to Kurtosis and RMS, and the next higher scores go to Peak to Peak, and Energy, while features like Skewness and Crest Factor contribute minimally. These insights are crucial for dimensionality reduction and optimizing model performance without sacrificing accuracy.

2) PARTIAL DEPENDENCE PLOTS (PDPS)

To visualize the marginal effect of individual features, PDPs were generated for all seven features; the comparative summary of the key observations and the practical insights is presented in Table 10.

TABLE 10
COMPARATIVE SUMMARY OF PDPS FOR MODEL EXPLAINABILITY

Feature	Key Observations	Practical Insight
RMS	Fluctuates below 0.3; stabilizes at ~0.645 after 0.3.	Critical for early fault detection; RMS < 0.3 is diagnostically significant.
Kurtosis	Drops around 4–5.5; sharp rise from ~6.5 to 9; plateaus after 9.	High kurtosis (>8) indicates confirmed fault; ideal for early warning systems.
Crest Factor	Declines until ~2.4; rises sharply between 2.4–2.9; saturates after 3.	Useful for detecting incipient defects; monitor values near 2.5 for onset of failure.
Impulse Factor	Flat response till ~2.5; rises steadily till ~3.5; saturates after.	Reflects impulsive faults (e.g., bearing cracks); most informative in mid-range.
Peak-to-Peak	Low and flat until 5; steep rise from 5–10; stable beyond 10.	Strong physical alignment; peak-to-peak > 6 should trigger alerts in maintenance systems.
Skewness	Low prediction around zero; score increases as skewness becomes more positive or negative; saturates >1.2	Monitoring skewness away from zero helps capture asymmetric fault behaviors like imbalance or misalignment.
Energy	The score rises sharply from 8 to 15; saturates after 15.	Energy is an excellent fault severity indicator; changes between 8–15 are most impactful.

The PDP analysis highlights that features like Kurtosis, Peak-to-Peak, Crest Factor, and Energy exhibit distinct threshold-based behavior, indicating clear decision boundaries within the model. Physically intuitive features such as RMS, Kurtosis, Energy, and Impulse Factor align well with domain knowledge, enhancing model trust. Notably, Kurtosis and Energy show sharp transitions and saturation, making them ideal for condition monitoring, while RMS and Impulse Factor offer high sensitivity in early fault detection. Skewness contributes unique asymmetry information, and Peak-to-Peak and Crest Factor respond strongly to impulsive events, making all seven features collectively valuable for robust and interpretable fault classification.

3) CORRELATION BETWEEN IMPORTANCE SCORE AND PDP

Table 11 presents the correlation between the importance scores sorted by mRMR and ReliefF algorithms and partial dependence plots for all seven features.

TABLE 11
CORRELATION BETWEEN FEATURE IMPORTANCE SCORES AND PARTIAL DEPENDENCE PLOTS (PDPS)

Feature	PDP Influence	mRMR Score Rank	ReliefF Score Rank	Correlation
Kurtosis	Strong threshold, high confidence region	1 (0.5179)	1 (0.0958)	Strong alignment; high PDP impact & statistical importance.
RMS	Sensitive at low range, stable at high	2 (0.4177)	2 (0.0774)	Excellent agreement; high PDP sensitivity & importance score.
Skewness	Nonlinear influence, both tails relevant	3 (0.4162)	6 (0.0191)	High in mRMR (global relevance), low in ReliefF (local variance); partial correlation.
Impulse Factor	Moderate, clear threshold zone (2.5–3.5)	4 (0.3252)	4 (0.0414)	Strong agreement; model captures informative behavior.
Peak-to-Peak	Sharp response in the 5–10 range	5 (0.2830)	3 (0.0734)	Medium agreement; higher importance in ReliefF aligns with PDP threshold behavior.
Crest Factor	Low–mid influence, sharp rise near 2.5–3	6 (0.2180)	5 (0.0221)	Moderate impact in PDP; both scores rank it lower consistent.
Energy	Strong monotonic influence (8–15)	7 (0.1660)	7 (0.0708)	Low mRMR score but PDP shows strong rise; model uses it well but it may be redundant with RMS.

From Table 11, the following conclusions can be drawn:

- **Feature Selection:** Kurtosis, RMS, and Impulse Factor are top candidates based on both model reliance (PDP) and statistical relevance (importance scores).
- **Redundancy Consideration:** Energy and RMS are likely correlated. mRMR penalizes this, but PDP shows Energy still drives decisions.
- **Model Explainability:** Features like Kurtosis and RMS improve explainability because their PDP behavior aligns with physical domain knowledge and feature selection metrics.

4) SENSITIVITY ANALYSIS VIA FEATURE PERTURBATION

A bidirectional sensitivity analysis was conducted by perturbing each feature by ±10% of its nominal value. The change in predicted class probabilities was recorded to quantify each feature's impact. The average absolute change was used as a measure of global sensitivity as shown in Fig. 15.

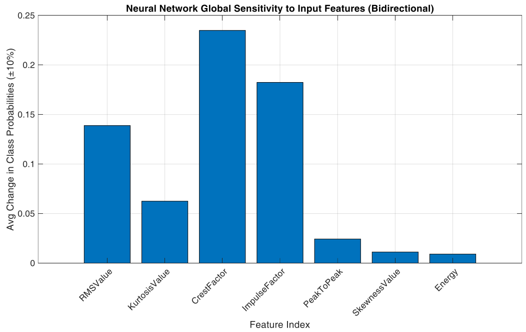


FIGURE 15. Neural Network Global Sensitivity to Input Features (Bidirectional).

This can be observed from Fig. 15 that the Crest Factor and Impulse Factor emerged as the most sensitive features, indicating a strong influence on the Wide Neural Network model's output. These features are directly related to sudden transient events and signal peaks, which are often associated with bearing faults.

RMS Value also demonstrated high sensitivity, reflecting the model's reliance on the overall energy content of the vibration signal.

Kurtosis Value showed moderate influence, aligning with its role in detecting impulsive characteristics often linked with localized defects.

Features like Peak-to-Peak, Skewness, and Energy exhibited minimal impact, suggesting that they provide redundant or less discriminative information for the neural network models.

5) CONFUSION MATRICES

The validation and test confusion matrices for the top-performing models such as the Ensemble Bagged Trees model, Fine Tree model, and Wide Neural Network model are presented in Fig. 16 through Fig. 18.

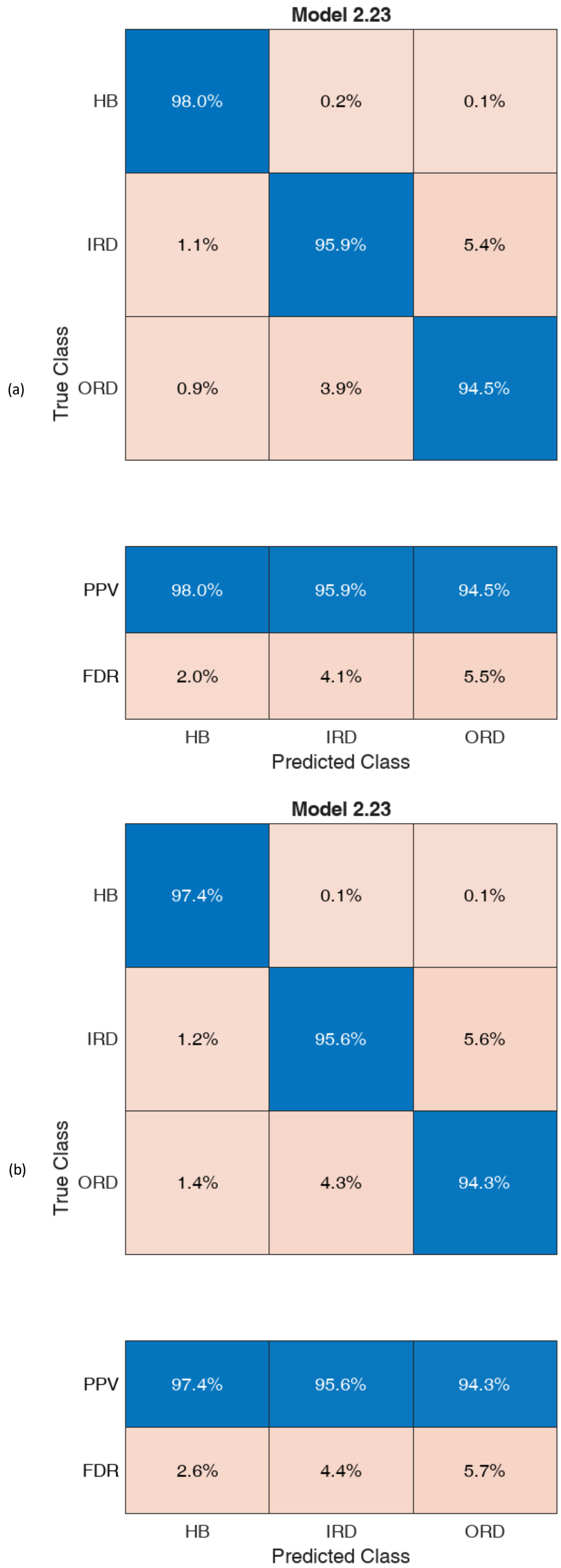


FIGURE 16. Confusion matrices for the Ensemble Bagged Trees model (a) Validation confusion matrix (b) Test confusion matrix.

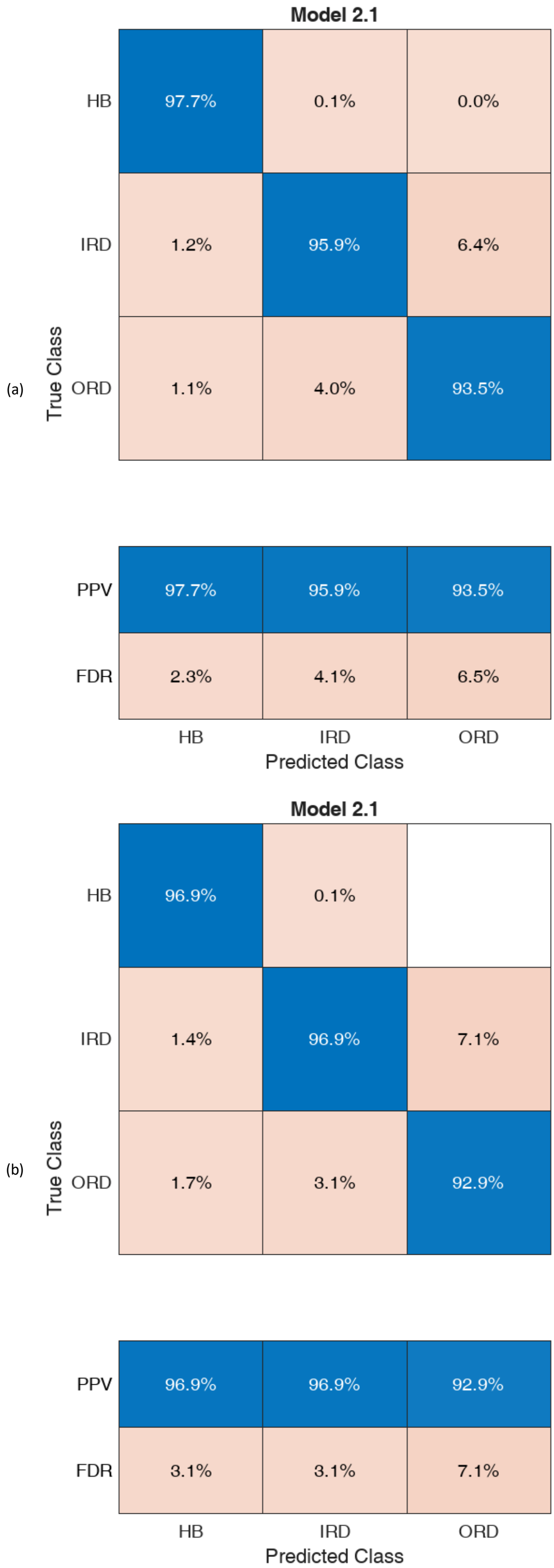


FIGURE 17. Confusion matrices for the Fine Tree model (a) Validation confusion matrix (b) Test confusion matrix.

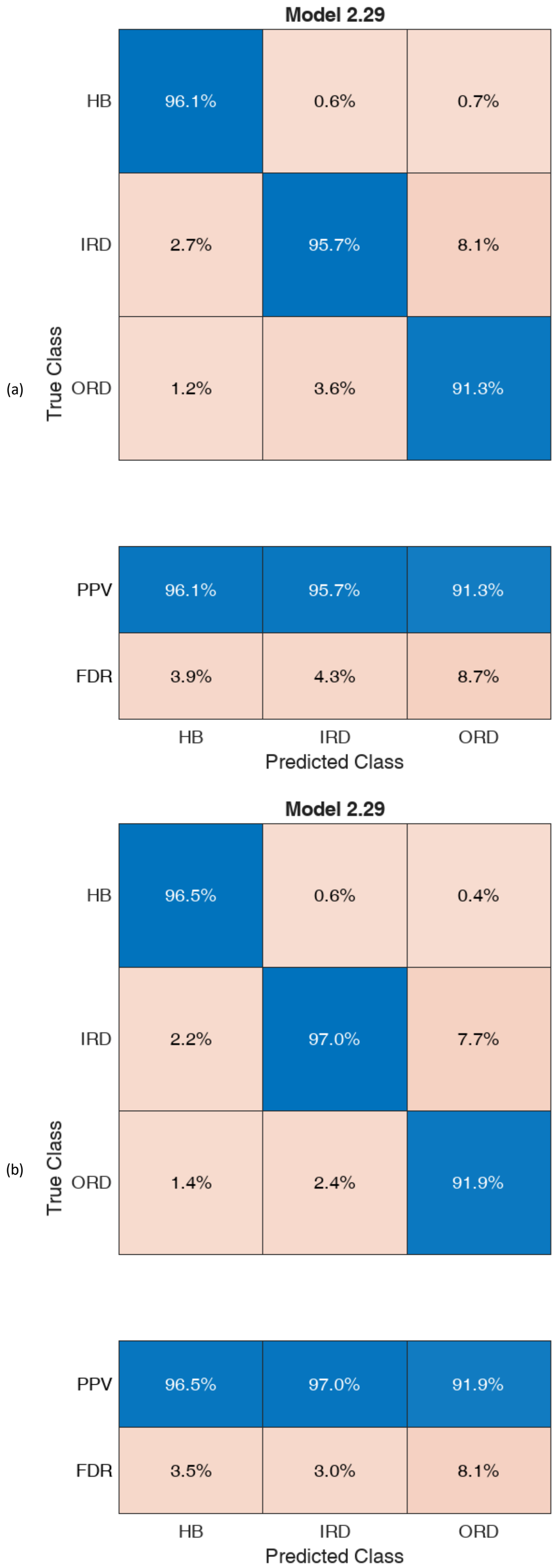


FIGURE 18. Confusion matrices for the Wide Neural Network model (a) Validation confusion matrix (b) Test confusion matrix.

The confusion matrices reveal that the models exhibit high classification accuracy and strong generalization capability. The minimum Positive Predictive Values (PPV) are 91.3% during validation and 91.9% during testing, while the False Discovery Rates (FDR) remain below 8.7% in validation and 8.1% in testing.

6) MAPPING OF THE PERFORMANCE METRICS WITH THE MODEL EXPLAINABILITY

TABLE 12
MAPPING OF PERFORMANCE METRICS TO EXPLAINABILITY

Metric	Explainability Insight
Validation Accuracy (%)	Gives an idea of how well the model generalizes on unseen data. Good starting point.
Validation Total Cost	This cost penalizes different types of misclassifications (e.g., missed faults more severely), and this directly reflects business risk. Very relevant for real-world deployment.
AUC of Validation ROC	Shows the model's ability to distinguish between classes. AUC closer to 1 implies high confidence.
Testing Accuracy (%)	Same as validation accuracy, but on the held-out test set — shows true performance under deployment conditions.
Testing Total Cost	Same as the validation cost. Strong practical relevance.
AUC of Testing ROC	Like validation AUC, can reinforce how confidently the model distinguishes class boundaries.
Training Time (s)	Does not aid interpretability directly, but shows computational efficiency, which matters in real-time/online systems.

Mapping performance metrics to model explainability reveals both diagnostic and deployment insights. Accuracy (validation and testing) indicates generalization and classification performance, while AUC reflects the model's confidence in distinguishing fault classes. Total cost metrics directly relate to real-world implications by penalizing critical misclassifications, enhancing the model's practical interpretability. Although training time lacks direct interpretability, it is crucial for evaluating computational feasibility in real-time applications.

IV. CONCLUSION

This deployment-oriented study benchmarks 33 machine learning algorithms for bearing fault classification using vibration data, focusing on practical applicability in real-world monitoring systems. Through three detailed case studies involving SKF7205, SKF7206, and SKF7207 bearings, and leveraging a robust dataset of 81,000 samples, the study identifies Ensemble Bagged Trees as the most reliable classifier across key performance metrics, including accuracy, AUC, and misclassification cost. Fine Tree models emerged as a strong alternative, offering a trade-off between performance and computational efficiency, while Wide Neural Networks showed high predictive power at the expense of training time. Feature selection through PCA and correlation analysis ensured a compact, non-redundant input space, enhancing model generalizability. These insights provide a practical reference for selecting and deploying machine learning models in condition monitoring systems for rotating machinery.

Limitation and Future Work: This study benchmarked 33 ML algorithms using seven time-domain statistical features due to their simplicity and low computational cost. However, this limited feature space may restrict model performance. Future work could incorporate frequency-domain (e.g., FFT, PSD) and time-frequency features (e.g., Wavelet Transform, EMD) to capture fault-related patterns more effectively.

Additionally, while this study focused on classical ML models, deep learning approaches and transfer learning offer promising avenues for automatic feature extraction and adaptability across varying conditions. Exploring these techniques could enhance the accuracy, robustness, and deployment potential of fault classification systems.

REFERENCES

[1] R. B. Randall and J. Antoni, “Rolling element bearing diagnostics—A tutorial,” *Mech. Syst. Signal Process.*, vol. 25, no. 2, pp. 485–520, Feb. 2011, doi: 10.1016/j.ymssp.2010.07.017.

[2] A. K. Jardine, D. Lin, and D. Banjevic, “A review on machinery diagnostics and prognostics implementing condition-based maintenance,” *Mech. Syst. Signal Process.*, vol. 20, no. 7, pp. 1483–1510, Oct. 2006, doi: 10.1016/j.ymssp.2005.09.012.

[3] Y. Lei, Z. He, and Y. Zi, “Application of an intelligent classification method to mechanical fault diagnosis,” *Expert Syst. Appl.*, vol. 36, no. 6, pp. 9941–9948, Aug. 2009, doi: 10.1016/j.eswa.2009.01.061.

[4] W. Zhang, G. Peng, C. Li, Y. Chen, and Z. Zhang, “A new deep learning model for fault diagnosis with imbalanced data,” *Knowl.-Based Syst.*, vol. 187, p. 104113, Jan. 2020, doi: 10.1016/j.knosys.2019.06.017.

[5] A. Althubaiti, F. Elasha, and J. A. Teixeira, “Fault diagnosis and health management of bearings in rotating equipment based on vibration analysis—a review,” Journal of Vibroengineering, vol. 24, no. 1, pp. 46–74, Feb. 2022.

[6] M. Elforjani and D. Mba, “Monitoring the wear of slow speed bearings using acoustic emission,” *Eng. Fract. Mech.*, vol. 77, no. 11, pp. 1778–1794, Jul. 2010, doi: 10.1016/j.engfracmech.2010.03.036.

[7] S. N. Patel and S. S. Dey, “Review of statistical features used in vibration analysis for bearing fault detection,” *Int. J. Eng. Technol.*, vol. 5, no. 3, pp. 1–4, 2013.

[8] T. Sugumaran, V. Ramachandran, and D. Ravikumar, “Fault diagnosis of roller bearing using fuzzy classifier and histogram features with focus on automatic rule learning,” *Expert Syst. Appl.*, vol. 38, no. 5, pp. 4901–4907, May 2011, doi: 10.1016/j.eswa.2010.09.114.

[9] L. Liang, Y. Zhang, and W. Y. Yan, “Fault detection using wavelet packet transform and artificial neural networks,” *Int. J. Comput. Commun. Control*, vol. 4, no. 3, pp. 309–318, Sep. 2009.

[10] P. K. Kankar, S. C. Sharma, and S. P. Harsha, “Fault diagnosis of ball bearings using machine learning methods,” *Expert Syst. Appl.*, vol. 38, no. 3, pp. 1876–1886, Mar. 2011, doi: 10.1016/j.eswa.2010.07.119.

[11] R. Parmar and A. Pandya, “Experimental study on effectiveness of SVM for classification of bearing faults,” *Procedia Eng.*, vol. 51, pp. 760–765, Jan. 2013, doi: 10.1016/j.proeng.2013.01.108.

[12] MathWorks, “Machine learning for predictive maintenance,” *MathWorks Documentation*, Accessed: May 2025. [Online]. Available: https://www.mathworks.com/solutions/predictive-maintenance.html

[13] X. Li, Q. Zhang, and Y. Ding, “An intelligent fault diagnosis method using RBF neural network and AR model,” *J. Vibroengineering*, vol. 17, no. 4, pp. 1651–1662, Jun. 2015.

[14] F. Fatima, M. R. Khan, and H. Rahman, “Machine learning-based diagnosis of rotor–bearing system faults using support vector machines,” *Measurement*, vol. 165, p. 108042, Jan. 2020, doi: 10.1016/j.measurement.2020.108042.

[15] Prasanta Kumar Samal, Sunil K., Imran M Jamadar, Srinidhi R, “AI-Enhanced Fault Diagnosis in Rolling Element Bearings: A Comprehensive Vibration Analysis Approach,” *FME Transactions*, vol. 52, no. 3, pp. 525–534, 2024.

[16] P. K. Samal, K. Sunil, I. M. Jamadar, M. Kowshik, and R. Srinidhi, “Fault Classification in Rolling Element Bearing Based on Vibration Signature Using Artificial Neural Network,” in *Proc. Int. Conf. on Robotics, Control, Automation and Artificial Intelligence*, Singapore: Springer Nature, 2023, pp. 521–533.

[17] Y. Liu et al., “RoBERTa: A Robustly Optimized BERT Pretraining Approach,” arXiv preprint arXiv:1907.11692, 2019.

[18] Cheruku R, Hussain K, Kavati I, Reddy AM, Reddy KS, “Sentiment classification with modified RoBERTa and recurrent neural networks,” Multimedia Tools and Applications, vol. 83, no.10, pp. 29399-417, Mar. 2024.

[19] Hema, M. S., R. Maheshprabhu, K. Sudheer Reddy, M. Nageswara Guptha, and V. Pandimurugan, "Prediction analysis for Parkinson disease using multiple feature selection & classification methods," *Multimedia Tools and Applications,* vol. 82, no. 27, pp. 42995-43012, 2023.

[20] E. D. Faust, R. Razavian, and J. N. Weiner, “Automated Parkinson's disease diagnosis using feature selection and machine learning,” *J. Biomed. Inform.*, vol. 100, p. 103324, Oct. 2019, doi: 10.1016/j.jbi.2019.103324.

[21] Rao, P. V., A. M. Reddy, K. S. Reddy, and J. L. Narayana, "Neural network aided optimized auto encoder and decoder for detection of COVID-19 and pneumonia using CT-scan," *J Theor Appl Inf Technol,* vol. 100, no. 21, pp. 6346-6360, 2022.

[22] S. Rehman, A. A. Khan, M. A. Gilani, and A. F. Moti, “COVID-19 detection using deep feature fusion and CNN,” *Comput. Biol. Med.*, vol. 132, p. 104389, Jan. 2021, doi: 10.1016/j.compbiomed.2021.104389.

[23] Sitharamulu, V., K. Rajendra Prasad, K. Sudheer Reddy, AV Krishna Prasad, and M. Venkat Dass, "Hybrid classifier model for big data by leveraging map reduce framework," *International Journal of Data Mining, Modelling and Management,* vol. 16, no. 1 pp. 23-48, 2024.

[24] S. Shobana and R. S. Moni, “Big data classification using hybrid machine learning model in healthcare,” *J. Ambient Intell. Humaniz. Comput.*, vol. 12, no. 4, pp. 4823–4832, Apr. 2021.

[25] S. Yerramaneni and S.K. Reddy, "A review on breast cancer detection using machine learning techniques", Int. J. Data Mining Modelling and Management, vol. 17, no. 2, 2025

[26] Khan, Saif Ur Rehman, Sohaib Asif, and Omair Bilal, "Ensemble Architecture of Vision Transformer and CNNs for Breast Cancer Tumor Detection From Mammograms," *International Journal of Imaging Systems and Technology,* vol. 35, no. 3, pp.e70090, 2025.

[27] Reddy, K. Sudheer, Niteesha Sharma, T. Ashalatha, and B. Ravi Raju, "An Intelligent Ensemble Architecture to Accurately Predict Housing Price for Smart Cities," In *International Conference on Intelligent Computing for Sustainable Development*, pp. 110-122. Cham: Springer Nature Switzerland, 2023.

[28] R. Chen, Y. Zhou, and Z. Zhou, “Ensemble learning for smart city infrastructure: A case study on traffic prediction,” *IEEE Access*, vol. 9, pp. 55443–55456, 2021, doi: 10.1109/ACCESS.2021.3071559.

[29] M. Wieland and M. Pittore, “Performance evaluation of machine learning algorithms for urban pattern recognition from multi-spectral satellite images,” *Remote Sensing*, vol. 6, no. 4, pp. 2912–2939, 2014.

[30] Belavagi, Manjula C., and Balachandra Muniyal, "Performance evaluation of supervised machine learning algorithms for intrusion detection," *Procedia Comput. Sci.*, vol. 89, pp. 117–123, 2016.

[31] S. Mittal and S. Tyagi, "Credit card fraud detection using machine learning: A comparative study," *Int. J. Inf. Technol.*, vol. 14, no. 2, pp. 575–582, Apr. 2022.

[32] R. B. W. Heng and M. J. M. Nor, "Statistical analysis of sound and vibration signals for monitoring rolling element bearing condition," Applied Acoustics, vol. 53, no. 1–3, pp. 211–226, Jan. 1998.

[33] Kurfess TR, Billington S, Liang SY, "Advanced diagnostic and prognostic techniques for rolling element bearings," Condition monitoring and control for intelligent manufacturing, pp. 137-65, 2006.

[34] T. B. N. and E. M. Igarashi, "A study on the prediction of abnormalities in rolling bearings," J JSLE Int, vol. 1, no. 7, pp. 1–6, 1980.

[35] Tandon N. "A comparison of some vibration parameters for the condition monitoring of rolling element bearings," Measurement, vol. 12, no. 3, pp. 285-289, 1994.

[36] Y. T. Su and S. J. Lin, "On initial fault detection of a tapered roller bearing: Frequency domain analysis," J Sound Vib, vol. 155, no. 1, pp. 75–84, 1992.

[37] Ying, L. U, "Decision tree methods: applications for classification and prediction," *Shanghai archives of psychiatry* 27, no. 2, pp. 130, 2015.

[38] Al-Aidaroos, Khadija Mohammad, Azuraliza Abu Bakar, and Zalinda Othman, "Naive Bayes variants in classification learning," In *2010 international conference on information retrieval & knowledge management (CAMP)*, pp. 276-281. IEEE, 2010.

[39] Vapnik, Vladimir N, "An overview of statistical learning theory," IEEE transactions on neural networks, vol. 10, no. 5, pp. 988-999, 1999.

[40] Lebold, Mitchell, Katherine McClintic, Robert Campbell, Carl Byington, and Kenneth Maynard, "Review of vibration analysis methods for gearbox diagnostics and prognostics," In Proceedings of the 54th meeting of the society for machinery failure prevention technology, vol. 634, p. 16, Virginia Beach, VA, 2000.

[41] Qian, Huimin, Yaobin Mao, Wenbo Xiang, and Zhiquan Wang, "Recognition of human activities using SVM multi-class classifier," *Pattern Recognition Letters,* vol. 31, no. 2, pp. 100–111, Jan. 2010.

[42] Duda, Richard O., and Peter E. Hart, "Pattern classification and scene analysis," *A Wiley-Interscience Publication* (1973).

[43] Wang, Jigang, Predrag Neskovic, and Leon N. Cooper. "Neighborhood size selection in the k-nearest-neighbor rule using statistical confidence." *Pattern Recognition*, vol. 39, no. 3, pp. 417–423, Mar. 2006.

[44] Domingos, Pedro, "A few useful things to know about machine learning," Communications of the ACM 55, no. 10, pp. 78-87, 2012.

[45] M.J.J. Douglass, "Book Review: Hands-on Machine Learning with Scikit Learn, Keras, and Tensorflow, 2nd edition by Aurélien Géron," Phys Eng Sci Med, vol. 43, no. 3, Sep. 2020.

[46] Rayjade, G., Bhagure, A., Kushare, P. B., Bhandare, R., Matsagar, V., & Chaudhari, A., "Performance evaluation of machine learning algorithms and impact of activation functions in artificial neural network classifier for bearing fault diagnosis," Journal of Vibration and Control, 2024.

PRASANTA KUMAR SAMAL is an assistant professor in the Department of Mechanical Engineering at The National Institute of Engineering, Mysuru, Karnataka, India. Currently, he is pursuing his Ph.D. He obtained his Master's degree from the Indian Institute of Science, Bangalore. He worked as a Deputy Manager in the Department of Computer Aided Engineering (CAE) at Mahindra & Mahindra Automotive and Farm Equipment Sector (AFS) at Mahindra Research Valley, Chennai, India. His research interests include vibration-based structural health monitoring, vibration-based condition monitoring of rotating machinery using machine learning, dynamic modeling and multi-body simulation of mechanical systems, finite element analyses, and experimental modal analysis.

PRAMOD KUMAR MALIK is an Assistant Professor in the School of Mechanical Engineering at Kalinga Institute of Industrial Technology (KIIT) Deemed University Bhubaneswar, Odisha, India. He completed his B. Tech. in the Mechanical Engineering Department at Veer Surendra Sai University of Technology (VSSUT) Burla. He obtained his Master's degree in Mechanical Engineering Department in Indian Institute of Science (IISc) Bangalore, Karnataka, India, and Ph.D. in Mechanical Engineering Department in Indian Institute of Technology (IIT) Bombay, Maharashtra, India. His research interests include Robotics, Kinematics and Dynamics of Machines, Tensegrity Mechanism, Mechanism Synthesis, Assistive Devices, Biomechanics, Deployable Structures, and Vibration. He is also the founder of Kurma Dynamics Private Limited, a Start-Up for designing and developing Underwater Autonomous Robots.

MANJUNATHA H J is a Lecturer in the Department of Mechanical Engineering at Government Polytechnic, Kushalnagar, Kodagu, Karnataka, In- dia. He earned his Bachelor's degree in Engineering from Malnad College of Engineering, Hassan, Karnataka, and his M. Tech from The National Institute of Engineering, Mysuru, Karnataka. Before joining academia, he worked as a Design Engineer at Quad Tooling Technology, Bengaluru. His research interests include vibration-based condition monitoring of rotating machinery using machine learning techniques.

Imran M Jamadar is an associate professor in the Department of Mechanical Engineering at The National Institute of Engineering, Mysuru, Karnataka, India. He obtained his doctoral degree from Sardar Vallabhbhai National Institute of Technology, Surat, India. His research interests include vibration-based condition monitoring of rotating machinery and vibration control. Dr. Imran M Jamadar has developed mathematical models for the prediction of the vibration amplitude from damaged bearings.

Srinidhi R is a Professor (Retd.), in the Department of Mechanical Engineering, JSS Science and Technology University (JSSSTU), Mysuru, Karnataka, India. His area of specialization is Machine Dynamics with a special focus on the development and testing of materials for acoustics and vibration applications. He obtained his doctoral degree from Kuvempu University, Karnataka, India, master's degree in Machine Dynamics from Indian Institute of Technology, Madras, Chennai, India