DOI: https://doi.org
Journal of Advanced Engineering & Applied Sciences
Journal homepage: https://rjsaonline.org/index.php/JAEAS

AI-Based Predictive Maintenance for Smart Manufacturing Systems: A Comparative Machine Learning Approach to Equipment Failure Prediction

Rayan Ahmed[1], Hira Aslam[2]
[1] Department of Electrical Engineering, Air University, Islamabad
[2] Department of Computer Engineering, Riphah International University
Email: rayanahmed002@gmail.com

ARTICLE INFO

Received:
April 15, 2026
Revised:
May 08, 2026
Accepted:
May 30 , 2026
Available Online:
June 10, 2026

Keywords:

predictive maintenance, artificial intelligence, machine learning, Random Forest, Support Vector Machine, XGBoost, feature importance, smart manufacturing

Corresponding Author:

rayanahmed002@gmail.com

ABSTRACT

Few problems in modern plants run up higher bills than unexpected downtime, which explains why predictive maintenance built on artificial intelligence (AI) has been getting so much attention. Using operational sensor readings gathered from factory machines, this study set up a quantitative comparison among several AI-driven predictive maintenance models. To forecast how likely a machine was to break down, the models drew on five predictors: vibration, temperature, pressure, total run time, and mechanical load. We trained three algorithms—Random Forest, Support Vector Machine, and Extreme Gradient Boosting (XGBoost)—and scored them on a held-out test split. We gauged model performance through accuracy, precision, recall, and F1-score, adding Root Mean Square Error where failure probabilities were estimated. XGBoost came out on top across every single metric, Random Forest trailed by only a small margin, and the Support Vector Machine suffered from distinctly lower recall even though its precision held up well. When we ran feature-importance checks on the XGBoost model, vibration amplitude and load stood out as the main drivers of breakdown, with run hours, temperature, and pressure playing lesser roles. What these results demonstrate is that feeding multivariate sensor inputs into ensemble learners produces dependable, practical failure warnings, giving plants a way off reactive repairs and fixed schedules and onto condition-based, predictive footing. The paper finishes by outlining what these results mean for plant managers, maintenance teams, and facilities working through Industry 4.0 upgrades, before pointing out avenues for follow-up work.

Introduction

Unscheduled equipment downtime eats into plant productivity and budgets more than almost anything else, and maintenance bills make up a large slice of total operating expenses across many facilities (Jardine et al., 2006). Traditional upkeep has long leaned on two basic models: reactive repairs carried out only post-breakdown, and preventive maintenance scheduled by the calendar regardless of real component wear. Each method comes with a clear downside: reactive fixes leave production vulnerable to sudden halts, while calendar-based servicing regularly wastes labor and parts on machinery that is running perfectly fine (Peng et al., 2010).

The rise of Industry 4.0 and smart factories has opened up fresh ways to handle maintenance decisions. Inexpensive sensors mounted directly on machinery can now log vibration, temperature, pressure, run time, and mechanical load around the clock, producing dense operational datasets that engineers simply did not have before (Lee et al., 2014). Condition-based maintenance relies directly on these data streams, initiating repairs from what the equipment shows in real time rather than sticking to a fixed calendar schedule (de Jonge & Scarf, 2020). Predictive maintenance takes that premise a step further: it runs statistical or machine learning models across historical sensor readings to estimate failure risks ahead of time, letting teams plan repairs both proactively and accurately (Zonta et al., 2020).

Predictive maintenance systems now rely heavily on machine learning and artificial intelligence as their core engine. When researchers first applied Support Vector Machines to condition monitoring, they proved that supervised algorithms could successfully read vibration and acoustic signals to classify how healthy a machine was (Widodo & Yang, 2007). More recent research indicates that ensemble algorithms like Random Forest and gradient-boosted methods such as XGBoost usually beat standalone classifiers and kernel techniques on multivariate sensor records, largely because they can capture intricate, nonlinear relationships across predictors (Carvalho et al., 2019; Xiang et al., 2021). Deep learning has also been brought into machine health monitoring, but it generally demands far larger training sets than tree-based ensembles do (Zhao et al., 2019).

Even with the expansion of literature on predictive maintenance powered by machine learning, notable blind spots persist. A lot of current work sticks to one model or tests only a few failure types, leaving factory teams with little practical guidance on which method fits their particular plant (Zhang et al., 2019). Few studies connect algorithm benchmarking with structured feature-importance evaluations that reveal which operating factors—vibration, temperature, pressure, running hours, or load—drive breakdown forecasts, even though that insight dictates how plants should budget for sensors and schedule servicing (Lei et al., 2020). Head-to-head assessments that evaluate Random Forest, Support Vector Machine, and XGBoost models under a shared set of metrics—namely accuracy, precision, recall, F1-score, and RMSE—are still scarce in predictive maintenance research (Susto et al., 2015).

To tackle those issues, this paper presents a quantitative benchmark of Random Forest, Support Vector Machine, and XGBoost algorithms for equipment failure prediction, evaluated against sensor readings typical of smart manufacturing plants. The models relied on vibration, temperature, pressure, operating hours, and machine load as predictors, with results measured through accuracy, precision, recall, F1-score, and RMSE. We then ran a feature-importance analysis to pinpoint which operating variables lined up most closely with equipment breakdowns. The rest of the paper is structured in four parts: the literature review summarizes previous studies on condition-based maintenance, machine learning diagnostics, and feature importance; the methodology outlines our sensor data, model training, and metrics; the analysis details the descriptive statistics, correlations, benchmark results, and feature rankings; and the final sections cover the discussion, conclusion, and recommendations, framing practical applications and future research paths.

Literature Review

Moving from Reactive to Predictive Maintenance

Condition-based maintenance developed to counter purely reactive fixes and calendar-based schedules, relying on live condition monitoring to guide decisions on the floor. In an early and widely cited review of machinery diagnostics and prognostics for condition-based maintenance, Jardine et al., (2006) laid out the core stages—data acquisition, data processing, and maintenance decision-making—that still underpin predictive systems today. Following this, Peng et al., (2010) surveyed machine prognostic models across four categories—physical, knowledge-based, data-driven, and hybrid—pointing out that cheaper, more pervasive sensors were pushing the field decisively toward data-driven methods. Looking at maintenance optimization models on a wider scale, De Jonge and Scarf (2020) observed that richer condition monitoring streams were steering researchers away from fixed replacement schedules and toward policies that adjust to actual machine health.

Smart Manufacturing and Industry

Predictive maintenance sits directly inside the Industry 4.0 agenda, which centers on wiring manufacturing facilities with networked sensors, big data analytics, and cyber-physical infrastructure. Lee et al., (2014) argued that smart analytics were indispensable for turning the flood of data from connected equipment into operational maintenance and manufacturing decisions, calling that conversion the core of Industry 4.0 and big

data production. Reviewing predictive maintenance research across the Industry 4.0 landscape, Zonta et al., (2020) found that diverse industrial sectors were increasingly pairing machine learning with sensor fusion and cloud analytics to spot impending failures in real time. Cheng et al., (2022) broadened this focus by evaluating visual analytics for predictive maintenance, reporting that anomaly detection and exploratory data analysis made up the bulk of what existing visualization systems actually handled.

Using Machine Learning Algorithms to Diagnose Faults and Predict Failures

A large share of the literature uses supervised machine learning algorithms to map sensor readouts to specific machine health conditions. In an early survey on Support Vector Machine use for machine condition monitoring and fault diagnosis, Widodo and Yang (2007) found that SVM classifiers delivered solid results on comparatively small, well-labeled datasets, though outcomes depended heavily on parameter tuning and kernel choice. For semiconductor fabrication, Susto et al., (2015) introduced a multi-classifier machine learning framework, showing that pairing models built for different forecasting horizons curbed both unexpected breakdowns and unneeded servicing compared to using a single classifier.

More recent predictive maintenance studies have zeroed in on ensemble learning techniques. In their systematic review of machine learning for predictive maintenance, Carvalho et al., (2019) found that Random Forest and gradient boosting were among the most common and effective tools, mainly because they handle noisy sensor data well and resolve nonlinear interactions between variables. Xiang et al., (2021) engineered an XGBoost fault diagnosis model for rolling bearings that extracted features from vibration signals, achieving classification scores higher than standard machine learning baselines. Gu et al., (2021) took a comparable path by tuning an XGBoost model with particle swarm optimization to pinpoint rotor faults, demonstrating that it beat several alternative intelligent classifiers in both diagnostic accuracy and computing speed. Pairing XGBoost-based classifiers with group-sparsity feature learning also proved effective for bearing fault diagnosis in work by Zhang et al. (2024), who noted that the setup held up better against the heavy background noise typical of factory floors.

Researchers have also turned to deep learning for monitoring machine health. Zhao et al.,(2019) surveyed deep learning in machine health monitoring—covering autoencoders, convolutional neural networks, and recurrent neural networks—and observed that while these architectures can excel on massive, high-dimensional sensor data, they usually demand more training samples and computing power than tree-based ensembles like Random Forest and XGBoost. In a broader survey of machine learning for fault diagnosis, Lei et al., (2020) pointed out that the field still lacks comparative studies testing different algorithm families on an equal footing—a gap the comparative setup of this study was built to fill.

Prognostics and Remaining Useful Life Prediction

Tied closely to failure classification is prognostics, the effort to estimate how much operating life a machine component has left. Lei et al., (2018) systematically surveyed machinery health prognostics across data collection, health indicator development, health stage segmentation, and remaining useful life estimation, emphasizing that the chosen indicators and feature engineering approaches largely dictated prognostic accuracy across the literature. Focusing specifically on machine tools, a review by Albertelli and Monno (2020) highlighted vibration and spindle load readings as two of the most reliable predictors of machine and tool wear across the surveyed research. Taken together, these studies suggest that the sensor metrics standard in prognostics—such as vibration, load, and run time—should also serve as strong predictors in failure classification problems like the one studied here.

Model **Interpretability and Feature Importance in Predictive Maintenance**

The growing complexity of machine learning models in predictive maintenance has driven a parallel emphasis on interpretability and feature-importance metrics. Zhang et al., (2019) reviewed data-driven predictive maintenance and stressed that pinpointing the most influential sensor signals matters not just for model transparency, but for helping manufacturers with tight budgets decide where to place sensors and allocate upkeep resources. Tree-based feature importance metrics, like Gini and permutation importance, see widespread use because they can be calculated straight from the fitted model without bringing in auxiliary post hoc explanation tools. Overall, the surveyed literature shows clear progress in applying Random Forest, Support Vector Machine, and XGBoost models to industrial fault diagnosis, yet surprisingly few papers run a

multi-metric comparison of all three alongside a structured feature-importance analysis of vibration, temperature, pressure, operating hours, and load—the exact gap this study was set up to address.

Methodology

Research Design

To examine how multivariate sensor readings relate to machinery failure, this study relied on a quantitative, predictive research design. Sensor readings were gathered to capture how an industrial fleet ran inside a smart manufacturing facility, recording vibration, temperature, pressure, operating hours, and machine load at set intervals alongside a yes-or-no flag for failures within a specified forecast window.

Variables and Data Sources

Sensor logs drawn from several industrial units yielded a core set of 10,000 machine-cycle records. Analysis relied on five distinct predictors: vibration amplitude (millimeters per second), operational temperature (degrees Celsius), pressure (bar), run-time accumulated since the preceding service check (operating hours), and current machine load expressed as a percentage of rated operating capacity. The outcome measure was binary: 1 marked an operational cycle that ended in failure within the following tracking interval; otherwise, it was scored 0. Breakdowns happened far less often than ordinary operating runs, matching the uneven class ratios standard in industrial diagnostic records.

Data Preprocessing

Before training any models, we screened the raw records for dropped telemetry values, background sensor noise, and uneven class frequencies. Where transmission dropouts common to shop-floor sensor networks caused missing readings, we reconstructed them through linear interpolation between neighbouring points on that same unit. Because tree-based learners ignore feature magnitude, inputs were kept in their original units for the Random Forest and XGBoost setups, whereas the Support Vector Machine required continuous variables to be centered at zero with unit variance. Because non-failure records vastly outnumbered breakdowns, we applied random oversampling to the failure cases in the training split so the classifiers would not simply settle on predicting normal runs every time. To make sure no information leaked across splits, we sorted the data by machine rather than by cycle, reserving 70% of observations for training, 15% for validation, and the final 15% as a held-out test block.

Model Development

To distinguish normal cycles from cycles that led to failure, three distinct machine learning models were built. For the Random Forest model, hyperparameters including tree count, tree depth, and the smallest allowable leaf population were selected through grid searches over the validation partition. A Support Vector Machine was fitted using a radial basis function kernel, with its penalty parameter and kernel coefficient determined by grid search on the validation data. Using the validation records, we calibrated the learning rate, booster rounds, tree depth, and regularization factors for an XGBoost model built on sequentially boosted shallow trees. To stop the XGBoost classifier from fitting to noise, we used early stopping pegged to its validation-set performance.

Model Evaluation

Model performance was judged on the held-out test set by deriving accuracy, precision, recall, and F1-score directly from the matrix of true versus predicted failure outcomes. Accuracy reflected the share of total observations that the model classified correctly. Precision captured the share of predicted failures that were true failures, whereas recall measured the share of true failures the model actually caught. Taking the harmonic mean of precision and recall yielded the F1-score, a single metric that balanced both properties to summarize classification performance. Each model also output a continuous failure probability, so we computed the Root Mean Square Error (RMSE) against the actual binary outcome as an extra, suitable check on predictive calibration.

Feature-Importance Analysis

Once models were built and evaluated, we evaluated feature importance through the top model's native Gini metric, checking it against a permutation procedure on that same model. To determine Gini importance, we

tallied the total drop in node impurity each predictor produced across all trees, then scaled the values so the five variables summed to 100%. We determined permutation importance by shuffling each predictor one by one and noting how much classification accuracy fell, which served to verify the Gini rankings. Predictors that caused steeper drops in impurity and larger accuracy losses when permuted were treated as having a greater impact on predicting machine failure.

Analysis

This section covers the sensor data's descriptive statistics, pairwise links between predictors and equipment failure, how the three models stacked up in classification, and the feature-importance findings.

Table 1

Descriptive Statistics of Sensor Variables (N = 10,000)

Variable	Mean	SD	Minimum	Maximum
Vibration (mm/s)	4.18	2.06	0.32	12.87
Temperature (°C)	68.3	11.4	38.6	104.2
Pressure (bar)	5.62	1.38	2.10	9.94
Operating Hours	812.4	437.9	12	1,998
Machine Load (%)	71.5	16.8	22.0	100.0

Note. SD = standard deviation. Operating hours were measured as cumulative hours since the last maintenance event.

Descriptive summaries for the sensor variables appear in Table 1, while Table 2 details how they correlate with equipment breakdowns. As Table 1 makes clear, the monitored operating conditions varied considerably across the board. Mean vibration stood at 4.18 mm/s with a standard deviation of 2.06, and temperature came in at an average of 68.3°C with an 11.4°C standard deviation. Operating hours spanned anywhere from 12 to 1,998 hours, reflecting major discrepancies in service schedules and run time. On average, machine load reached 71.5%, which points to equipment generally running fairly close to full capacity. Looking at the correlations in Table 2, vibration tied most directly to breakdowns ($r = .68$, $p < .001$), with machine load ($r = .59$, $p < .001$) and operating hours ($r = .51$, $p < .001$) trailing just behind. Temperature correlated moderately with failure ($r = .37$, $p < .001$), while pressure yielded the lowest correlation overall ($r = .24$, $p < .001$). Because every variable showed a positive, statistically significant correlation, it appears that heavier workloads and prolonged use increase the odds of a breakdown. Because vibration had the strongest correlation, irregular mechanical shaking looks like a valuable early signal that an equipment defect is setting in.

Table 2

Point-Biserial Correlation Coefficients Between Predictor Variables and Equipment Failure

Predictor Variable	r	p
Vibration	.68	< .001
Machine Load	.59	< .001

Predictor Variable	r	p
Operating Hours	.51	< .001
Temperature	.37	< .001
Pressure	.24	< .001

Note. N = 10,000. All correlations were statistically significant at the .001 level.

Table 3

Classification Performance of the Machine Learning Models on the Test Subset

Model	Accuracy	Precision	Recall	F1-Score	RMSE
Random Forest	.941	.912	.887	.899	.187
Support Vector Machine	.903	.896	.814	.853	.241
XGBoost	.958	.934	.921	.927	.152

Note. RMSE was calculated between each model's predicted failure probability and the actual binary failure outcome. Higher accuracy, precision, recall, and F1-score values and lower RMSE values indicate superior predictive performance.

Table 4

Feature Importance Analysis: Relative Contribution of Operational Variables in the XGBoost Model

Variable	Relative Importance (%)	Rank
Vibration	32.6	1
Machine Load	26.8	2
Operating Hours	19.4	3
Temperature	13.1	4
Pressure	8.1	5

Note. Relative importance values were derived from Gini importance and cross-validated using permutation importance. Values were normalized to sum to 100%.

The predictive performance of the machine-learning models and how much each operational variable contributed to forecasting equipment failure are shown in Tables 3 and 4. Across all three models assessed, XGBoost performed best overall, yielding an accuracy of .958, precision of .934, recall of .921, and an F1-score of .927, alongside a study-low RMSE of .152. Random Forest put up solid numbers too, but the Support Vector Machine lagged behind with lower classification metrics. Because overlooked breakdowns bring unplanned downtime and drive up repair costs, XGBoost's high recall is a major asset for predictive maintenance, showing it caught failing units especially well. Looking at feature importance, vibration had the largest impact

at 32.6% of the model's relative weight, with machine load coming next at 26.8% and operating hours at 19.4%. Temperature accounted for 13.1%, whereas pressure registered the smallest contribution at 8.1%. These outcomes match the correlation analysis, which also identified vibration and machine load as having the strongest ties to equipment failure. Taken together, the findings imply that tracking vibration, regulating machine load, and timing maintenance around operating hours could be especially valuable when designing practical predictive-maintenance programs. In the end, XGBoost delivered both dependable classification accuracy and clear guidance on which operating parameters track most closely with machine failure.

Table 5

Comparison of Predicted and Actual Failure Status for a Sample of Test Machines (XGBoost Model)

Machine ID	Actual Status	Predicted Status	Predicted Failure Probability
M-014	Failure	Failure	.91
M-027	No Failure	No Failure	.08
M-035	Failure	Failure	.84
M-041	No Failure	No Failure	.14
M-058	Failure	No Failure	.46
M-062	No Failure	No Failure	.19

Note. Predicted status was assigned using a probability threshold of .50. Machine M-058 represents a false negative case in which the predicted failure probability fell below the classification threshold.

How the XGBoost model performs in practice is shown in Table 5 across a sample of test machines. Five of the six machines shown were classified correctly by the model, split between three failure and two no-failure cases. M-014 and M-035 were accurately flagged as failures with predicted probabilities of .91 and .84, whereas M-027 and M-041 were correctly marked as no-failure cases alongside relatively low predicted probabilities of .08 and .14. Machine M-062 was also correctly classified as having no failure, with a predicted probability of .19. Machine M-058 turned out to be a false negative: the machine actually failed, but its predicted probability was .46, just under the .50 cutoff. This case illustrates that despite the model's high general predictive accuracy, particular failure events can still be tough to flag. On an actual shop floor, keeping false negatives down matters a great deal; an undetected failure can bring unexpected downtime, maintenance expenses, and broken operations. Tracking probabilities directly and tuning the threshold appropriately could make the predictive-maintenance model considerably more useful in practice.

Discussion

Among the three models evaluated, XGBoost delivered the most accurate and best-calibrated failure predictions, leading across accuracy, precision, recall, and F1-score while posting the lowest RMSE. That outcome squares with earlier comparative work showing gradient boosting methods outperforming both single-tree and kernel-based classifiers on industrial fault diagnosis (Xiang et al., 2021; Gu et al., 2021; Zhang et al., 2024). Random Forest also performed solidly, trailing XGBoost by only a small margin on every metric; this lends weight to the broader finding by Carvalho et al. (2019) that tree ensembles cope well with the noisy, multivariate sensor feeds common to predictive maintenance. Precision for the Support Vector Machine was competitive, yet its recall fell short of both ensemble methods; this pointed to a higher rate of missed failures, fitting Widodo and Yang's (2007) point that SVM performance hinges on parameter choices and may not hold up as well as ensembles against noisy operational data in higher dimensions.

Identifying vibration and machine load as the primary drivers of failure lines up well with the prognostics literature discussed above. Reviews by Lei et al. (2018) on machinery health prognostics and Albertelli and

Monno (2020) on machine tool management drew much the same conclusion, highlighting vibration and load-related readings as primary markers of mechanical wear. Pressure played a noticeably smaller role here, which implies that under these specific operating conditions, pressure swings had less direct bearing on failure than mechanical stress and load did, though that might not hold for machines where hydraulic or pneumatic lines drive the breakdown.

The correlation values pointed in the same direction, showing vibration and machine load to have the strongest individual links to equipment failure. That matches what the wider literature on condition-based maintenance reports, namely that signs of mechanical stress tend to give the earliest and most dependable warning that failure is coming (Jardine et al., 2006; Peng et al., 2010). The single false negative that turned up when checking predictions against true failure outcomes points to a wider issue raised by Zhang et al. (2019): strong predictive maintenance models can still misjudge borderline cases, so human technicians still need to keep an eye on automated systems.

Conclusion and Recommendations

In this study, we ran a quantitative comparison of Random Forest, Support Vector Machine, and XGBoost models to predict industrial equipment failures from sensor streams tracking vibration, temperature, pressure, operating hours, and load. XGBoost scored highest on accuracy and calibration across all measures, with Random Forest a close second, whereas the Support Vector Machine had noticeably lower recall despite matching precision. According to the feature-importance analysis, vibration amplitude and machine load drove equipment failure more than any other operating factors, with operating hours, temperature, and pressure following in that order. Taken together, the results show that ensemble models, especially gradient boosted approaches like XGBoost, can anchor reliable sensor-based predictive maintenance in smart factories, helping plants move away from reactive or calendar-driven repairs toward predictive, condition-based upkeep. Plants should prioritize fitting and calibrating vibration and load sensors on their critical assets, given how heavily those metrics drove the predictions, and treat XGBoost or similar gradient boosting models as the standard choice for multi-sensor failure classification, leaving Support Vector Machines for smaller, well-curated data pools. Maintenance teams should also maintain human supervision over borderline calls, especially in light of the false-negative risk documented here, and pair any predictive software with explicit escalation steps whenever a failure probability sits near the decision threshold. Subsequent work should test these methods across multiple operational plants with broader mixes of machinery and failure types, combine remaining useful life estimates with standard binary classifications, and measure the real economic and workflow effects of running these models live rather than relying strictly on retrospective performance checks.

References

1. Albertelli, P., & Monno, M. (2020). A review of prognostics and health management of machine tools. The International Journal of Advanced Manufacturing Technology, 107(5–6), 2843–2863. https://doi.org/10.1007/s00170-020-05202-3

2. Carvalho, T. P., Soares, F. A. A. M. N., Vita, R., Francisco, R. P., Basto, J. P., & Alcalá, S. G. S. (2019). A systematic literature review of machine learning methods applied to predictive maintenance. Computers & Industrial Engineering, 137, Article 106024. https://doi.org/10.1016/j.cie.2019.106024

3. Cheng, X., Chaw, J. K., Goh, K. M., Ting, T. T., Sahrani, S., Ahmad, M. N., Kadir, R. A., & Ang, M. C. (2022). Systematic literature review on visual analytics of predictive maintenance in the manufacturing industry. Sensors, 22(17), Article 6321. https://doi.org/10.3390/s22176321

4. de Jonge, B., & Scarf, P. A. (2020). A review on maintenance optimization. European Journal of Operational Research, 285(3), 805–824. https://doi.org/10.1016/j.ejor.2019.09.047

5. Gu, K., Wang, J., Qian, H., & Su, X. (2021). Study on intelligent diagnosis of rotor fault causes with the PSO-XGBoost algorithm. Mathematical Problems in Engineering, 2021, Article 9963146. https://doi.org/10.1155/2021/9963146

6. Jardine, A. K. S., Lin, D., & Banjevic, D. (2006). A review on machinery diagnostics and prognostics implementing condition-based maintenance. Mechanical Systems and Signal Processing, 20(7), 1483–1510. https://doi.org/10.1016/j.ymssp.2005.09.012

7. Lee, J., Kao, H.-A., & Yang, S. (2014). Service innovation and smart analytics for Industry 4.0 and big data environment. Procedia CIRP, 16, 3–8. https://doi.org/10.1016/j.procir.2014.02.001

8. Lei, Y., Li, N., Guo, L., Li, N., Yan, T., & Lin, J. (2018). Machinery health prognostics: A systematic review from data acquisition to RUL prediction. Mechanical Systems and Signal Processing, 104, 799–834. https://doi.org/10.1016/j.ymssp.2017.11.016

9. Lei, Y., Yang, B., Jiang, X., Jia, F., Li, N., & Nandi, A. K. (2020). Applications of machine learning to machine fault diagnosis: A review and roadmap. Mechanical Systems and Signal Processing, 138, Article 106587. https://doi.org/10.1016/j.ymssp.2019.106587

10. Peng, Y., Dong, M., & Zuo, M. J. (2010). Current status of machine prognostics in condition-based maintenance: A review. The International Journal of Advanced Manufacturing Technology, 50(1–4), 297–313. https://doi.org/10.1007/s00170-009-2482-0

11. Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2015). Machine learning for predictive maintenance: A multiple classifier approach. IEEE Transactions on Industrial Informatics, 11(3), 812–820. https://doi.org/10.1109/TII.2014.2349359

12. Widodo, A., & Yang, B.-S. (2007). Support vector machine in machine condition monitoring and fault diagnosis. Mechanical Systems and Signal Processing, 21(6), 2560–2574. https://doi.org/10.1016/j.ymssp.2006.12.007

13. Xiang, C., Ren, Z., Shi, P., Zhao, H., & Wei, C. (2021). Data-driven fault diagnosis for rolling bearing based on DIT-FFT and XGBoost. Complexity, 2021, Article 4941966. https://doi.org/10.1155/2021/4941966

14. Zhang, J., Zhang, M., Wang, D., & Yang, M. (2024). Bearing fault classification diagnosis based on group-sparsity learning and AVOA-XGBoost. Journal of Vibration and Shock, 43(18), 96–105.

15. Zhang, W., Yang, D., & Wang, H. (2019). Data-driven methods for predictive maintenance of industrial equipment: A survey. IEEE Systems Journal, 13(3), 2213–2227. https://doi.org/10.1109/JSYST.2019.2905565

16. Zhao, R., Yan, R., Chen, Z., Mao, K., Wang, P., & Gao, R. X. (2019). Deep learning and its applications to machine health monitoring. Mechanical Systems and Signal Processing, 115, 213–237. https://doi.org/10.1016/j.ymssp.2018.05.050

17. Zonta, T., da Costa, C. A., da Rosa Righi, R., de Lima, M. J., da Trindade, E. S., & Li, G. P. (2020). Predictive maintenance in the Industry 4.0: A systematic literature review. Computers & Industrial Engineering, 150, Article 106889. https://doi.org/10.1016/j.cie.2020.106889



2026 by the authors; Journal of Advanced Engineering & Applied Sciences (JAEAS). This is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC-BY) license (http://creativecommons.org/licenses/by/4.0/).