# From Preventive to Predictive: Advancing Timely and Accurate Aircraft Prognostics

Philippa Scroggins and Sidi Lu
Department of Computer Science, William & Mary, Williamsburg, VA 23185, USA
prscroggins@wm.edu, sidi@wm.edu

***Abstract*—The aviation industry heavily relies on effective maintenance strategies to ensure operational safety, but routine inspections under conventional preventive maintenance approaches often lead to high costs, delays, and cancellations. Predictive maintenance offers a transformative alternative by forecasting equipment failures, optimizing schedules, and enhancing safety. In this study, we propose and evaluate three novel variants of the Gradient-Boosting Regression Tree algorithm for the prediction of aircraft turbofan engine's Remaining Useful Life (RUL). Our research investigates how feature engineering, operating conditions, and fault modes affect predictive performance across diverse engine operating scenarios. We conduct a comprehensive comparative analysis of our proposed algorithms against seven state-of-the-art methods, including both traditional machine learning, deep learning, and hybrid approaches. Additionally, we introduce the Margin-Adjusted Reliability Score (MARS), a novel benchmarking metric that incorporates both prediction accuracy and timeliness, addressing gaps in existing evaluation methods. By providing insights into algorithm interpretability and performance, this work contributes to the development of efficient, transparent, and industry-relevant predictive maintenance solutions, advancing the state of fault prognostic systems in aviation. The datasets, tools, and algorithms from this work will be open-sourced to support community research.**

***Index Terms*—Aircraft, turbofan engine, predictive maintenance, edge computing, fault forecasting, time series data**

## I. Introduction

Commercial aviation is essential to global mobility. Between 1970 and 2010, passenger numbers increased more than eightfold [1]. Today, the Federal Aviation Administration (**FAA**)'s Air Traffic Organization manages over 45,000 flights daily, transporting 2.9 million passengers across 29 million square miles of airspace [2]. These operations rely heavily on efficient maintenance, repair, and overhaul (MRO) processes to ensure safety, reliability, and cost-effectiveness [3].

Although maintenance-related aircraft failures are rare, even minor lapses can lead to devastating consequences. Recent incidents, such as the Jeju Air crash in Korea, which resulted in 179 fatalities and two injuries, underscore the critical importance of robust maintenance practices in ensuring aviation safety and reliability [4]. Economic pressures and fluctuating passenger demand further compel airlines to continually refine their MRO strategies to enhance safety, address operational demands, and control costs. This work aims to address these challenges by enabling accurate and timely prognostics for commercial aviation, as presented in Fig. 1.

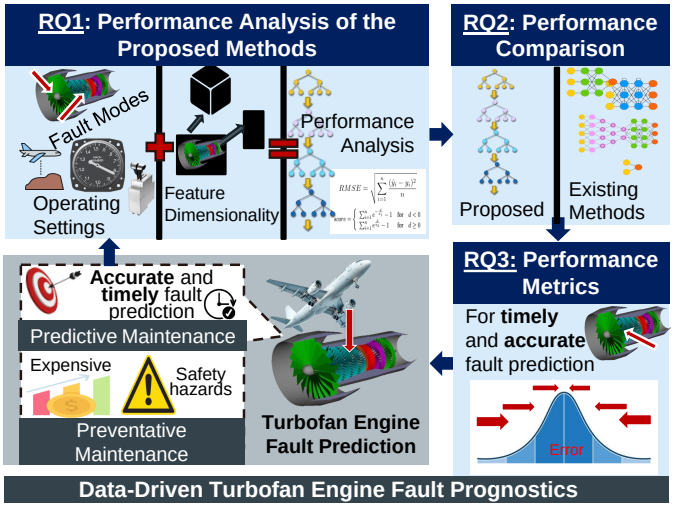



Fig. 1. An overview of our experimental procedures for turbofan engine fault prognostics. We address three key research questions (**RQs**): performance analysis of our proposed method (top-left), comparative performance evaluation (top-right), and proposal of a new performance metric (bottom-right). **RQ1** investigates how the performance of our proposed RUL prediction algorithm changes with varying complexities of operating scenarios as well as with different feature engineering techniques. **RQ2** benchmarks the performance of our proposed method against other methods in similar studies. **RQ3** examines existing performance metrics and proposes a new metric tailored to better address the specific challenges of the turbofan engine prognostics task.

### A. Aircraft Turbofan Engines

Aircraft turbofan engines are a cornerstone of modern commercial aviation, delivering high thrust while maintaining fuel efficiency by combining elements of turbojet and turboprop designs [5]. Turbojets generate thrust by compressing air, mixing it with fuel, and igniting the mixture to produce high-velocity exhaust gases [6]. While effective at high speeds, they lack fuel efficiency at lower subsonic speeds. In contrast, turboprops, which combine a jet engine with a propeller powered by the turbine, excel in fuel efficiency at speeds below 500 miles per hour [7].

To address design limitations at high subsonic speeds, turbofan engines split incoming air into two streams: one directed into the core for thrust and the other bypassing the core to reduce noise and improve fuel efficiency [8], as shown in Fig. 2. The bypass ratio, the proportion of air bypassing the core, is key to their design. Modern turbofans, like the geared turbofan (**GTF**), achieve bypass ratios up to 12:1, balancing fuel efficiency and performance across various speeds. This blend of engineering and practicality makes them essential for

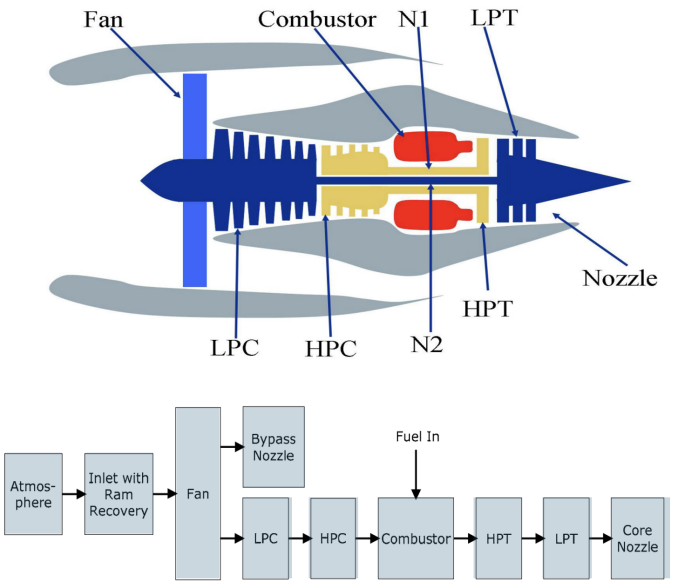



Fig. 2. A simplified diagram of an aircraft turbofan engine. The top section illustrates the locations of its five rotating components: the fan, Low-Pressure Compressor (**LPC**), High-Pressure Compressor (**HPC**), High-Pressure Turbine (**HPT**), and Low-Pressure Turbine (**LPT**). It also marks the positions of the combustor and nozzle, as well as the metrics "fan spool speed" (**N1**) and "core spool speed" (**N2**). The bottom section provides a visual representation of the engine's operational process [9].

cost-effective, environmentally sustainable aviation.

### *B. Aircraft Turbofan Engine Maintenance*

To ensure safety in the high-stakes aircraft industry, where errors are minimally tolerated, regularly scheduled engine maintenance has long been the standard practice. However, this traditional approach is costly, as maintenance is performed regardless of the engine's actual condition. While preventative maintenance aims to reduce the risk of engine failure, it can inadvertently introduce new mechanical issues during routine procedures, resulting in expensive disruptions. For instance, a single flight cancellation can cost an airline approximately $140,000, while each hour of delay incurs expenses of about $17,000 [10]. These financial pressures underscore the pressing need for more efficient maintenance strategies.

Advancements in edge computing [11], the Industrial Internet of Things (**IIoT**) [12], and Prognostics and Health Management (**PHM**) [13] have paved the way for predictive maintenance strategies, which enhance engine uptime, cost efficiency, and safety. Predictive maintenance relies on continuous monitoring of engine parameters, such as temperature and vibration, using advanced sensors. Combined with off-board operational data, these parameters are analyzed by machine learning models to detect anomalies and assess degradation levels [14]. This process enables optimized maintenance scheduling. For example, predictive maintenance strategies have allowed the Boeing 787 to reduce flight delays and cancellations by 30% and unscheduled engine removals by 20%, significantly improving operational efficiency [15].

Despite its promise, implementing predictive maintenance for aircraft engines poses several challenges. Data ownership disputes between engine manufacturers and commercial operators, stringent regulatory requirements by the FAA, and the technical limitations of predictive models all complicate adoption [15]. Additionally, effective machine learning models for prognostics require failure data to ensure predictive accuracy and reliability. However, obtaining such data would require allowing engines to fail, a prospect incompatible with aviation's stringent safety standards [16], [17].

To address the scarcity of real-world failure data, researchers have developed sophisticated simulation tools to generate synthetic datasets for developing and testing prognostic models. Initiatives such as the PHM Data Challenge competitions[1] and NASA's Prognostics Data Repository[2] have significantly advanced PHM research by providing realistic simulation datasets and benchmarks for evaluating these developed algorithms.

### *C. Datasets and the Need for Standardized Evaluation Metrics*

The National Aeronautics and Space Administration (**NASA**) Prognostics Data Repository offers datasets for researchers to compare and evaluate prognostic algorithms. Among these are datasets from the Commercial Modular Aero-Propulsion System Simulation (**C-MAPSS**) tool [9], which models turbofan engine operations and introduces faults with varying degradation levels. The first C-MAPSS dataset was followed by an extended release of four sub-datasets with varying complexities [18]. These datasets have become essential benchmarks for prognostic algorithm research.

Despite the widespread use of these datasets, comparing results across studies remains difficult due to unclear result computation methods, leading to inconsistencies. The need for a standardized evaluation metric to assess these approaches is increasingly critical.

### *D. Research Questions and Contributions*

**Research Questions**. Motivated by the challenges outlined above, this study utilizes four sub-datasets from the Turbofan Engine Degradation Simulation dataset, which simulates run-to-failure trajectories for a small fleet of turbofan engines under realistic flight conditions.

To provide clear insights into decision-making processes for engine maintenance and foster trust in safety-critical aviation domains, this study developed three variants of Gradient-Boosting Regression Trees (**GBRT**) as tree-based algorithms instead of relying on computation-intensive deep learning methods. Tree-based methods offer interpretability, computational efficiency, and robustness on small to medium-sized datasets. In contrast, deep learning models often require extensive tuning and lack transparency. Previous studies in time-series data prediction have shown that, with advanced feature engineering, tree-based algorithms can achieve competitive accuracy while avoiding the complexity and resource demands

[1]https://data.phmsociety.org/

[2]https://www.nasa.gov/intelligent-systems-division/discovery-and-systems-health/pcoe/pcoe-data-set-repository/

of deep learning [19], making them a practical and accessible baseline for this study.

To be concrete, our research addresses three primary research questions (**RQs**), as shown in Fig. 1:

($i$) How effectively can a tree-based prognostics algorithm predict the Remaining Useful Life (**RUL**) of simulated turbofan engines, considering varying complexities of engine operating conditions and the influence of feature engineering techniques (**RQ1**)?

($ii$) How does the tree-based algorithm's accuracy compare to existing methods, and how does this performance change when accounting for prediction timeliness (**RQ2**)?

($iii$) Can we design an improved performance metric for RUL prediction models that incorporates both accuracy and timeliness, provides clear and distinct evaluations of these factors, and enables standardized comparisons across datasets and solutions (**RQ3**)?

**Contributions of This Work**. From a technical perspective, we develop three variants of the GBRT algorithm for predicting the Remaining Useful Life (**RUL**) of turbofan engines. We evaluate its performance, compare it against seven state-of-the-art (**SOTA**) methods, including both traditional machine learning and deep learning approaches, recently published or widely used for time-series data prediction, and introduce a new performance benchmarking metric. *The datasets, tools, and algorithms developed in this work will be made open-source to support and advance related research within the community.* Our specific contributions are as follows:

- We propose and develop accurate and efficient GBRT-based prognostic solutions to estimate the RUL of turbofan engines (Section IV).
- For **RQ1**, we evaluate how the algorithm's prediction accuracy and timeliness are influenced by different feature dimensionalities applied to the data, engine operating conditions, and degradation fault modes. We identify strategies to optimize the predictive performance with these factors in mind (Section IV).
- For **RQ2**, we compare the prediction accuracy and timeliness of our method with other published solutions, including ConvGAT [20], FCDAE-CNN-LSTM [21], Support Vector Machine (**SVM**) [22], Multi-Layer Perceptron (**MLP**) [22], Deep Belief Network (**DBN**) [22], Long Short-Term Memory (**LSTM**) [23], and Deep Convolutional Neural Network (**DCNN**) [24] (Section V).
- For **RQ3**, we propose a novel performance metric that transparently measures both accuracy and timeliness of RUL predictions. This metric provides a clear and standardized way to compare different methods. (Section VI).
- We provide an extensive discussion of our experimental observations for **RQ1**, **RQ2**, and **RQ3**, offering in-depth explanations, contributing to the broader knowledge base for enhanced aircraft safety (Sec. VII).

The rest of this paper is organized as follows: Sec. II reviews related work. The experiment datasets and hardware configuration are detailed in Sec. III. Extensive experimental results of RQ1, RQ2, and RQ3 are shown in Sec. IV, Sec. V, and Sec VI, respectively. We present our discussion points in Sec. VII, and Sec. VIII concludes the entire paper.

## II. Related Work

### A. Remaining Useful Life forecasting

*1) Importance of RUL Forecasting in PHM:* RUL forecasting is an important step within the PHM process. RUL prediction forecasts the time remaining until a component or system reaches the end of its functional lifespan [25]–[27]. Having the knowledge of a system's RUL value allows for proactive maintenance planning, reduces the risk of equipment failures, minimizes maintenance and supply chain costs, and optimizes operational schedules.

*2) General Solution Types for RUL Prediction:* ① **Model-Based *vs.* Data-Driven Methods**. Approaches to RUL forecasting are categorized as model-based or data-driven methods, though there are many that include characteristics of both. Model-based approaches rely on physical laws or analytical models of the system under study. Data-driven models use machine learning or statistical techniques to learn patterns of system or component degradation, often from time-series sensor data, which vary in computational complexity and accuracy depending on the quality and quantity of the data required for the task domain [28].

② **Simulation-Based Data for RUL Prediction**. Because producing run-to-failure data is limited in a practical setting, simulation-produced datasets are often used to inform the development of data-driven RUL forecasting methods. These datasets simulate degradation scenarios for a specific physical system or component. Some examples include the degradation of bearings, lithium-ion batteries, composite materials, and milling machines [29]. Data usually includes detailed sensor readings over time that portray failure trajectories for these systems so that data-driven models can be trained to predict the degradation level of the system. One example of a recent solution is [30], which predicts the RUL of rolling bearings by integrating feature extraction through Synchrosqueezing Wavelet Transform (**SSWT**) and Random Projection (**RP**) with a deep learning architecture that combines Residual Networks (**ResNet**) and temporal attention layers. Another example is [31], which predicts the RUL of lithium ion batteries using a GM-PFF model that combines grey modeling with a particle flow filter.

*3) Representative RUL Prediction Methods:* The challenge of predicting the RUL of turbofan engines has been a popular topic of study since the release of the C-MAPSS simulated datasets. There are several categories of solutions identified in this literature [32], illustrated below.

① **Health index-based methods**. Health index-based models map sensor measurements to a health index for each training unit, which is then linked to RUL. Recent work in this category has focused on detecting system degradation using deep learning models. For example, [33] combines deep belief networks with self-organizing map neural networks to

build a health index that captures correlations between multi-component systems, significantly improving RUL prediction.

② **Similarity-based matching**. Similarity-based matching methods create a library of system instances with known failure times. For a test instance, similarity with library instances is evaluated to estimate and aggregate RUL. Recent advancements include integrating autoencoder architectures and failure mode-specific metrics to enhance RUL prediction accuracy. For example, [34] uses a classifier to identify the failure mode and guide RUL prediction.

③ **Neural network-based approaches.** Perhaps the most prominent category, especially as of recently, are neural network-based methods. These methods transform engine trajectory data into a multidimensional feature space, using corresponding RUL values to label feature vectors. Supervised learning is then applied to map feature vectors to RUL. Recent work in this category includes a multi-dimensional attention mechanism combined with a feature-sequence dimensional convolution network, which captures interactions in feature dimensions and temporal sequences, improving RUL prediction accuracy on datasets like NASA's turbofan engine data and XJTU-SY [35]. Hybrid deep learning models, such as Convolutional Long Short-Term Memory (**CNN-LSTM**) [36] [27], and FCDAE-CNN-LSTM [21], have also gained popularity.

④ **Emerging Hybrid Solutions**. The most recent solutions, however, combine one or more of these categories to address each different parts of the problem. For instance, [37] integrates Temporal Convolutional Networks (TCNs) for temporal feature extraction with a Bi-LSTM to learn salient temporal patterns. [38] uses a parallel prognostic network to discern degradation features for RUL prediction, and also incorporates Monte Carlo dropout to produce a probabilistic prediction, addressing predictive uncertainty within the solution. [39] uses feature squeeze excitation (**FSE**) to assign weights to sensors, discerns degradation information using LSTM augmented with a softmax temporal permutation selecting (**STPS**) mechanism, and employs fully connected networks (**FCNs**) to map features to RUL values.

### B. Research Gaps in Previous Work

*1) Lack of Interpretability:* Despite significant advancements in the field, critical gaps and challenges persist. One major issue is the lack of interpretability in high-performing RUL prediction models. Many models are highly complex, resulting in increased computational demands and reduced transparency, both of which are critical for safety-critical applications. Black-box models, while often accurate, fail to provide the insights needed for integration into modern PHM systems, which aim to combine diagnostics, controls, and multi-objective optimization in real time [40]. Interpretability and transparency are essential for integrating RUL forecasting into broader PHM frameworks, enabling reliable maintenance recommendations, improved mission readiness, and reduced operating costs.

*2) Computational Complexity:* Another significant challenge is computational complexity. Many advanced methods, especially hybrid neural network models, rely on computationally intensive feature extraction and overall high processing demands. These requirements often make such models impractical for real-time or resource-constrained environments [25]. While these methods achieve impressive accuracy, their computational overhead makes them unsuitable for applications that require a balance between accuracy, efficiency, and transparency.

*3) Inadequate Evaluation Metrics:* Evaluation and comparison of RUL prediction methods also face considerable gaps. Widely used metrics such as Root Mean Squared Error ((Eq. 3) and the PHM Score (Eq. 2) fail to provide a comprehensive assessment of model performance. While accuracy is an important measure, it is insufficient for selecting PHM models that must perform well across multiple dimensions. Prognostic Performance Indicators (PPIs), which evaluate various aspects of prognostic approaches, need to be defined to establish a structured evaluation framework [41]. Although the 2008 PHM Challenge metric [18] offers additional insights beyond accuracy, it still falls short of enabling standardized and thorough comparisons across methods.

## III. Tool and Data Overview

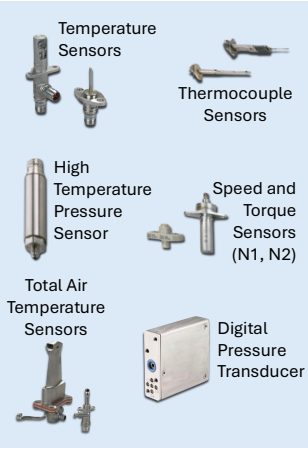
### A. Experiment Tools

C-MAPSS is a MATLAB and Simulink-based tool that simulates a large commercial turbofan engine in the 90,000 lb thrust class. It models engine operations across various conditions, including altitudes from sea level to 40,000 feet, Mach numbers from 0 to 0.90, and sea-level temperatures from -60 to 103 degrees Fahrenheit. The tool also includes a power management system for simulating engine performance under different thrust levels and flight conditions [9]. In this work, the full list of sensor measurements in the C-MAPSS simulated data is presented in Fig. 3.

### B. Turbofan Engine Degradation Simulation Dataset

Using C-MAPSS, Saxena, Goebel, Simon, and Eklund created the Turbofan Engine Degradation Simulation dataset [18]. The tool was employed to generate sensor response surfaces and operability margins for engine components as functions of flow and efficiency. Each engine simulation began with an initial deterioration level, with an exponential rate of flow and efficiency loss applied to simulate a fault with progressive effects. Fault directions and progressions were imposed randomly, with a time-dependent health index tracking degradation.

To introduce realism, measurement noise was added to the simulated data, which otherwise assumes "ideal" sensors and actuators with no dynamics, delays, errors, or biases [9]. The resulting dataset includes time-series sensor data capturing engine degradation from normal operation to failure. Each dataset reflects the usage history of a fleet of engines, supporting the development of algorithms for predicting RUL.




| Sensor Measurement |
|---|
| Fan Inlet Temperature (◦R) |
| LPC Outlet Temperature (◦R) |
| HPC Outlet Temperature (◦R) |
| LPT Outlet Temperature (◦R) |
| Fan Inlet Pressure (psia) |
| Bypass-Duct Pressure (psia) |
| HPC Outlet Pressure (psia) |
| Physical Fan Speed (rpm) |
| Physical Core Speed (rpm) |
| Engine Pressure Ratio (P50/P2) |
| HPC Outlet Static Pressure (psia) |
| Ratio of Fuel Flow to Ps30 (pps/psia) |
| Corrected Fan Speed (rpm) |
| Corrected Core Speed (rpm) |
| Bypass Ratio |
| Burner Fuel-Air Ratio |
| Bleed Enthalpy |
| Required Fan Speed |
| Required Fan Conversion Speed |
| High-Pressure Turbines Cool Air Flow |
| Low-Pressure Turbines Cool Air Flow |

Fig. 3. Overview of the sensor measurements represented in the C-MAPSS simulated data. Different types of engine sensors [42] are pictured on the left, and to the right is the full list of sensor measurements included in the C-MAPSS generated dataset. Note that LPC stands for low-pressure compressor, HPC stands for high-pressure compressor, and LPT stands for low-pressure turbine. Different types of engine sensors are pictured on the top left.

### C. Dataset Structure

As shown in Fig. 3, the datasets in this work represent a fleet of turbofan engines. Each simulated engine's data is structured as an $n \times 26$-dimensional matrix where $n$ represents the engine cycles per trajectory. Each row contains data for one cycle: the first column identifies the engine, the second indicates the cycle number, columns 2-5 denote operational settings, and columns 6-26 record engine sensor measurements. The dataset contains four sub-datasets, each simulating different fault modes, and operating conditions, detailed in Table I.

| | FD001 | FD002 | FD003 | FD004 |
|---|---|---|---|---|
| **Training Trajectories** | 100 | 260 | 100 | 249 |
| **Testing Trajectories** | 100 | 259 | 100 | 248 |
| **Operating Conditions** | 1 | 6 | 1 | 6 |
| **Fault Modes** | HPC | HPC | HPC & Fan | HPC & Fan |

TABLE I
DATASET DETAILS. NUMBER OF TRAINING AND TESTING TRAJECTORIES, OPERATING CONDITIONS, AND FAULT MODES IN EACH OF THE FD001, FD002, FD003, AND FD004 SUB-DATASETS.

### D. Engine Operating Condition Complexity

Engine operability margins, which indicate the engine's distance from operational limits (e.g. stall and temperature limits), vary with operational conditions [18]. The sub-datasets model different levels of operating condition complexity, which are determined by the combination of altitude, mach number, and throttle-resolver angle (**TRA**) parameters.

- FD001 and FD003: Simpler operating conditions.
- FD002 and FD004: More complex operating conditions.

FD001 and FD003 represent more simple operating scenarios, while FD002 and FD004 simulate complex operating conditions. Sensor data frequency distributions, shown in Fig. 4, reveal that the simpler operating scenarios (FD001 and FD003) often produce Gaussian sensor value distributions while more complex datasets (FD002 and FD004) exhibit a disparate distribution of sensor values. *This indicates that patterns of degradation will differ with operating conditions, so the predictive model must be capable of distinguishing these nuanced patterns*. Each sub-dataset is split into separate training and testing datasets. In each training set, engine data ends when the health index reaches 0. In each testing set, the rows of data truncate before the engine reaches the point of failure. The goal is to predict the RUL of the turbofan engine.

## IV. RQ1: Proposed Method and Performance Analysis

To address **RQ1**, we explain our design of the proposed prediction framework and examine its performance forecasting turbofan engine RUL across the various engine operating conditions in the Turbofan Engine Degradation Simulation dataset. We examine how different feature selection approaches impact performance, and we identify the approach that yields the best results across each operating scenario. Fig. 5 provides an overview of our experimental setup.

### A. Experiment Setup

*1) Data Preparation:* ① **Piecewise RUL target function**. The ground truth RUL values in the testing set are provided only for the final engine cycle, while the training set does not contain RUL labels. Without access to a physics-based model, we applied a piecewise linear degradation function (Fig. 6) as utilized in prior research [43], [44], to cap RUL values and account for non-linear degradation beyond certain usage thresholds. This approach mitigates the risk of overestimating RUL and more accurately captures the actual degradation patterns observed in turbofan engines.

② **Data normalization**. Variations in operating conditions cause significant discrepancies in sensor measurement patterns, making data normalization essential to ensure consistent feature scaling. The feature data is normalized using Eq. 1:

$$X_i' = \frac{X_i - \mu}{\gamma} \tag{1}$$

Here, $X_i$ represents the original sensor measurement for a specific feature, $X_i^{'}$ denotes the normalized value of $X_i$, $\mu$ indicates the mean value of the feature across all samples in the dataset, and $\gamma$ refers to the standard deviation of the feature across all samples. This normalization process scales the data to have a mean of 0 and a standard deviation of 1, ensuring that features with varying units or magnitudes are brought to a consistent scale. This improves both the performance and stability of machine learning models.

*2) Distinction of Operating Conditions:* Given the influence of engine operating conditions on the degradation patterns captured in sensor data, we used K-means clustering to analyze the variations relative to these conditions [45]. The clustering process was guided by the three operational parameters (columns 3-5 in the dataset): Altitude (**OPS1**), Mach Number (**OPS2**), and Throttle Revolver Angle (**OPS3**). This process segments the dataset into distinct clusters representing unique combinations of operating conditions. Sub-datasets FD001 and FD003 contain limited variation in these

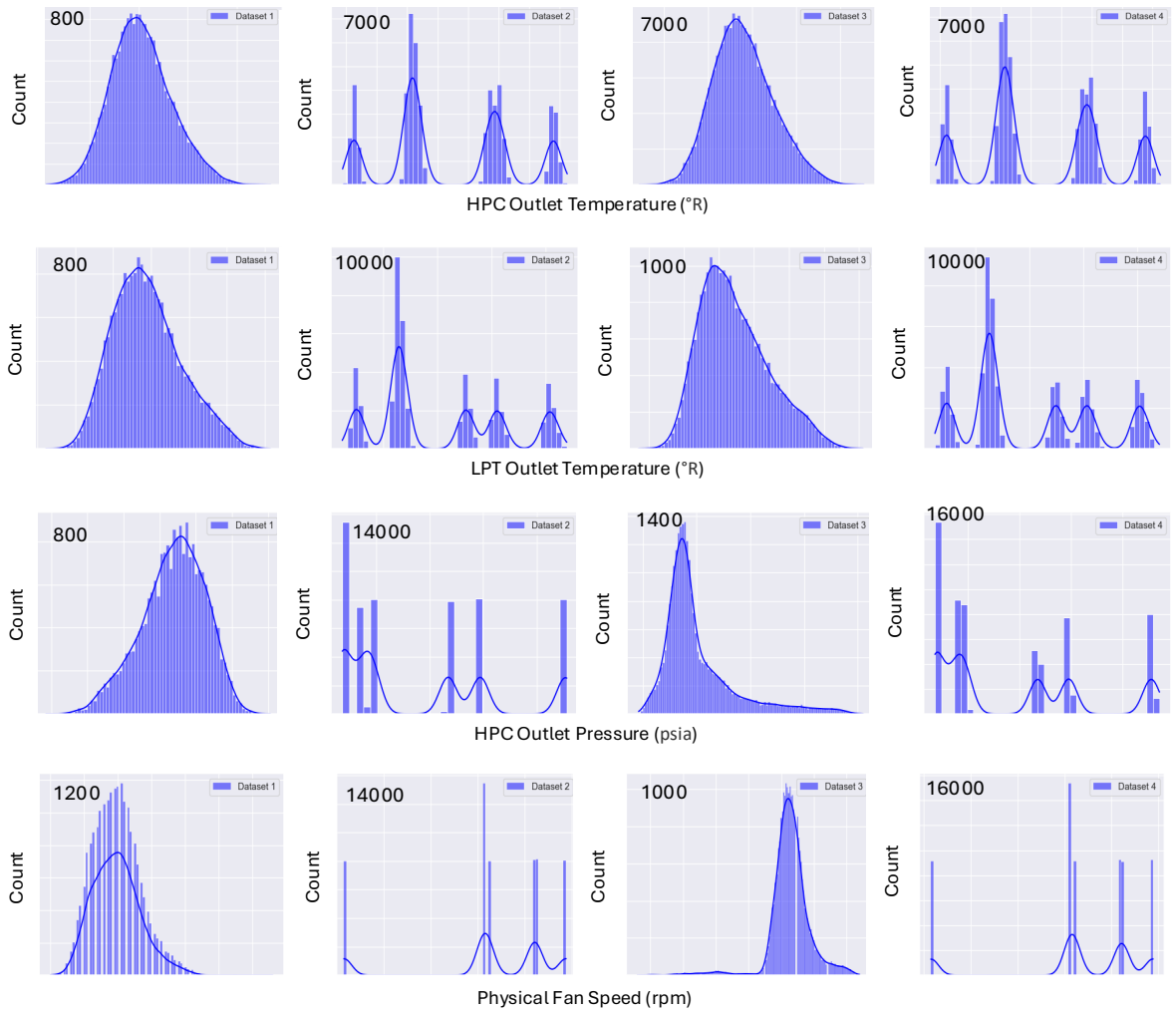



Fig. 4. Sensor value distributions for sensors, top to bottom: HPC Outlet Temperature, measured in degrees Rankine; LPT Outlet Temperature, measured in degrees Rankine; HPC Outlet Pressure, measured in absolute pressure (pounds per square inch); Physical Fan Speed, measured in rotations per minute.

parameters and thus produce a single cluster, while sub-datasets FD002 and FD004 contain more variety and accordingly produce six distinct clusters. These clusters were incorporated into the dataset as re-engineered features, labeled OP1 through OP6, to signify the presence of specific operating conditions for each sensor measurement. These features are encoded as boolean values, where a value of 1 indicates that a sensor measurement belongs to a particular cluster, and 0 indicates otherwise.

*3) Feature Engineering:* To reduce data dimensionality while preserving variance, we applied Principal Component Analysis (**PCA**) preserving 95% variance to produce a separate feature-selected dataset. To capture more subtle and intricate relationships within sensor measurement patterns, we employed polynomial feature mapping to create the separate, expanded version of the dataset. We chose to experiment with different dimensionalities in order to gain insight into how best to account for the variation in sensor measurement patterns and correlations across operating scenarios.

### *B. Proposed Methodology*

In this work, we propose three novel variants of the GBDT algorithm. GBDT is an ensemble learning method that constructs a sequence of decision trees, where each tree learns from the residual errors of its predecessors to minimize prediction error. Training begins with a base tree and sequentially adds trees to correct prior errors, using a learning rate to prevent overfitting. The final model, a weighted combination of all trees, effectively captures complex nonlinear relationships between engine sensor measurements and RUL values [46].

We chose GBDT as the foundational framework not only for its low computational requirements but also for its intuitive predictive transparency, which provides valuable insights into the factors influencing its predictions. Unlike deep learning methods, often regarded as black-box models, GBDT provides a transparent structure that clearly demonstrates the impact of individual features on its output.

Specifically, we develop three variants of the GBRT algorithm: **GBRT I**, trained on the original dataset without feature engineering; **GBRT II**, trained on the feature-selected version of the dataset; and **GBRT III**, trained on the feature-mapped

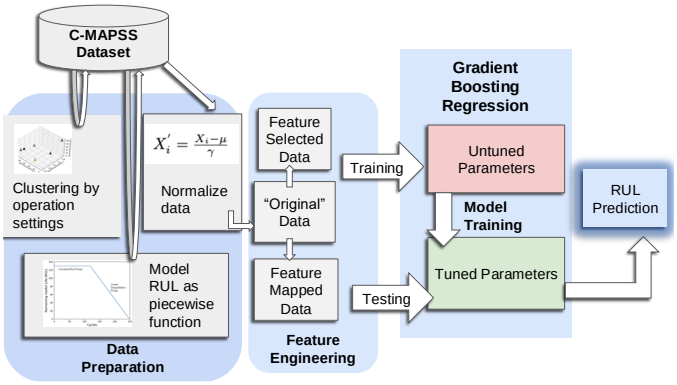



Fig. 5. Experiment setup and training procedure for the proposed Gradient Boosting Regression Tree-based RUL prediction framework. Data is initially clustered based on operating settings to define operating conditions, which are incorporated as new features. A piecewise function is used to model ground truth RUL values for non-target sensor measurement rows, and values are added as an additional feature. The data is then normalized to reduce the variance of sensor measurement patterns across different operating scenarios. Following this, feature selection and polynomial feature mapping are applied to generate two alternative versions of the dataset. Training data for each version is used to fine-tune a set of GBRT parameters. Once optimized, the tuned model predicts RUL values on the corresponding test dataset.

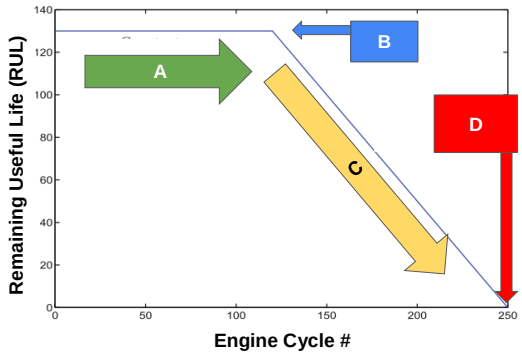



Fig. 6. Piecewise engine degradation function. In segment A, the RUL is constant, reflecting a stable health index of the modeled engine. At point B, the onset of a fault is represented, which initiates the degradation process. As the fault progresses in segment C, the declining RUL represents the diminishing health index of the engine. At point D, the engine's RUL reaches zero, indicating complete degradation.

version of the dataset. Each version underwent tuning on the training set of each sub-dataset to optimize several architecture hyperparameters, including the learning rate, number of boosting stages, maximum tree depth, minimum samples required to split a node, minimum samples per leaf, number of features considered for the best split, and the fraction of samples used for training base learners. Following this tuning process, each model version was tested on the corresponding testing set to assess performance.

### C. Evaluation Metrics

The development of uniform evaluation metrics remains challenging for this task due to the diverse needs of the aviation industry and evolving regulatory standards. Unfortunately, the lack of a standardized performance metric complicates the comparison of results across studies and the evaluation of the progress within this field [32]. We thus adopted the scoring metric recommended by the dataset developers in [18], referred to in this paper as the "PHM Score" (Equation 2). This metric, which was used to assess submitted solutions to the 2008 PHM Challenge [18], penalizes late failure predictions more heavily than early ones, reflecting the aerospace industry's emphasis on early risk aversion.

Despite its design, the PHM Score has several limitations. It is sensitive to outliers, biased towards algorithms that underestimate RUL, and produces a single numeric value that can be too ambiguous for comprehensive performance comparisons. To counter these limitations, we incorporate an additional evaluation metric based on the frameworks proposed in [47] and [48]. Among these, Root Mean Squared Error (**RMSE**), defined in Eq. 3, was selected for its ability to equally penalize early and late RUL forecasts. RMSE is widely used in related studies, allowing direct comparisons [20]–[24], [49], [50].

$$\text{PHM Score} = \begin{cases} \sum_{i=1}^{n} e^{-\frac{d}{13}} - 1 & \text{for} \quad d < 0 \\ \sum_{i=1}^{n} e^{\frac{d}{10}} - 1 & \text{for} \quad d \geq 0 \end{cases} \tag{2}$$

$n =$ number of engine cycles

$d =$ predicted RUL - true RUL

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{d^2}{n}} \tag{3}$$

$n =$ number of predictions

$d =$ predicted RUL - true RUL

The PHM Score aggregates these penalties across all cycles, but its sensitivity to outliers and lack of interpretability limit its effectiveness in some cases. RMSE offers a clear and interpretable measure of prediction accuracy, making it a valuable addition to our evaluation framework.

### D. Experiment Results

Figure 7 illustrates the performance of the three GBRT variants (GBRT I, GBRT II, GBRT III) across the four sub-datasets which each represent a unique operating scenario. The results emphasize the significant impact of operating condition complexity on predictive performance.

In the simplest scenario, sub-dataset FD001, which simulates a single fault mode and operating condition, GBRT I (trained on the original dataset) achieves both the best RMSE and PHM Score. Conversely, GBRT II (trained on the feature-selected data) produces the weakest accuracy and PHM.

For FD003, which also simulates a single operating condition but two fault modes, GBRT I achieves the lowest RMSE, slightly outperforming GBRT III. However, GBRT III achieves a significantly better PHM Score, indicating an improved prediction timeline for the sub-dataset's degradation pattern.

Sub-dataset FD002, which includes six operating conditions, simulates more complex degradation patterns. In this case, GBRT III achieves both a slightly better RMSE and a superior PHM Score , which indicates the advantage of using feature mapping to learn complex, non-linear degradation patterns.

Finally, in the most intricate scenario, FD004, which combines six operating conditions with two fault modes, GBRT I

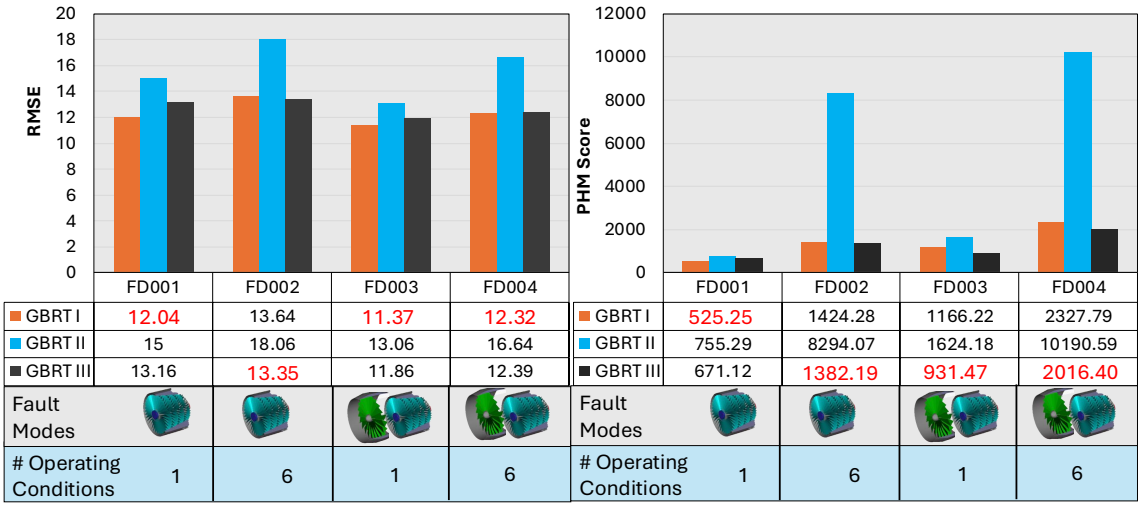



Fig. 7. The RMSE for each algorithm variant (GBRT I: original features, GBRT II: feature-selected data, GBRT III: feature-mapped data) is shown on the left, while the corresponding PHM Score values are displayed on the right. Fault modes (HPC degradation and combined HPC and fan degradation) and the number of operating conditions simulated in each sub-dataset are indicated between panels below both graphs. The best RMSE and PHM Score values are highlighted in red.

achieves the lowest RMSE. However, GBRT III again achieves the best PHM Score.

## V. RQ2: Performance Comparison

### A. Experiment Design

To benchmark the performance of our proposed GBRT-based algorithms, we compared GBRT I and GBRT III against several recently published solutions and several widely-implemented techniques for turbofan engine RUL prediction. We chose the recent methods, ConvGAT [20], FCDAE-CNN-LSTM [21] based on their novel contributions to this field, and the rest were chosen for their frequent adoption in related studies, making them suitable benchmarks for assessing the relative strengths and weaknesses of our approach.

All selected methods were trained and tested on each of the four sub-datasets using the same evaluation metrics: RMSE and PHM Score. This consistent experimental setup ensures this performance comparison is consistent and insightful.

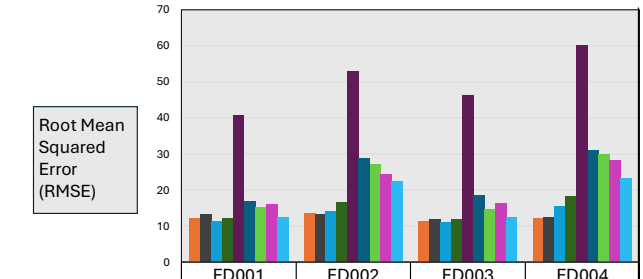
For further insight, we provide a review of the comparison solutions' architectures:

- **ConvGAT [20]:** The ConvGAT solution combines graph neural networks (GNNs) with sensor embeddings to learn complex, non-linear relationships between sensor measurements, which flexibly models spatial correlations between individual sensor data patterns. This architecture also includes a convolutional layer before the GNN, which extracts features from sensor data to serve as initial node feature vectors.
- **FCDAE-CNN-LSTM [21]:** This solution integrates a Fully Convolutional Denoising Autoencoder (FCDAE) with a combined CNN-LSTM architecture. The FCDE combines a fully convolutional network (FCN) for feature reconstruction with a denoising autoencoder (**DAE**) for noise reduction, ensuring minimal reconstruction error during data preprocessing. The parallel CNN-LSTM architecture is employed to capture both spatial and temporal characteristics of the denoised data.
- **Other methods [22]:** We also provide a comparison to [22]'s implementation of traditional machine learning models such as SVM, Multi-Layer Perceptrons (**MLP**), and standalone Deep-Belief Networks (**DBNs**). We also make a comparison to the performance of [51]'s implementation of LSTM and [52]'s implementation of DCNN. While these architectures are foundational, with the advancement of new hybrid techniques, their performance often serves as a baseline for constructing and evaluating more complex techniques.

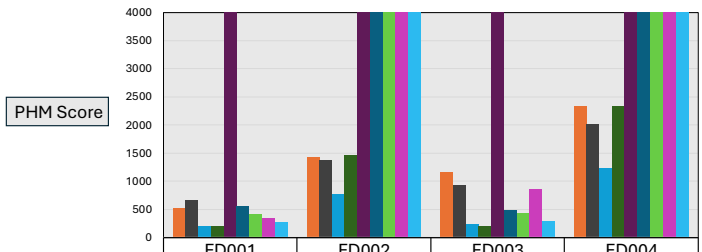
### B. Experiment Results

Figure 8 displays the results of this comparison. Both GBRT I and GBRT III demonstrate strong performance when measured by RMSE: GBRT III achieves the lowest RMSE of 13.4 on FD002, while GBRT I achieves the lowest RMSE of 12.3 on FD004. On FD001 and FD003, ConvGAT achieves the best RMSEs of 11.3 and 11.0, respectively [20]. However, GBRT I delivers comparable results with RMSEs of 12.0 for FD001 and 11.4 for FD003. These results show that GBRT I is able to achieve a highly competitive level of accuracy. Additionally, unlike other methods that exhibit notable accuracy variations between the simpler operating scenarios (FD001/FD003) and more complex ones (FD002/FD004), both GBRT I and GBRT III maintain their consistent accuracy across all datasets, demonstrating their adaptability to diverse degradation patterns.

Turning to the comparison of PHM Scores, while neither GBRT method achieves the best scores overall, GBRT III performs competitively on FD004 and FD002. For FD004, GBRT III achieves a score of 2016.4, second only to ConvGAT's 1231.17 [20]. Similarly, on FD002, both GBRT I and




| | FD001 | FD002 | FD003 | FD004 |
|---|---|---|---|---|
| GBRT I (Proposed) | 12 | 13.6 | 11.4 | 12.3 |
| GBRT III (Proposed) | 13.2 | 13.4 | 11.9 | 12.4 |
| ConvGAT | 11.3 | 14.1 | 11.0 | 15.5 |
| FCDAE-CNN-LSTM | 12 | 16.6 | 11.8 | 18.2 |
| SVM | 40.7 | 53 | 46.3 | 60 |
| MLP | 16.8 | 28.8 | 18.5 | 31 |
| DBN | 15.2 | 27.1 | 14.7 | 29.9 |
| LSTM | 16.1 | 24.5 | 16.2 | 28.1 |
| DCNN | 12.6 | 22.4 | 12.6 | 23.3 |




| | FD001 | FD002 | FD003 | FD004 |
|---|---|---|---|---|
| GBRT I (Proposed) | 525.25 | 1424.28 | 1166.22 | 2327.79 |
| GBRT III (Proposed) | 671.12 | 1382.19 | 931.47 | 2016.4 |
| ConvGAT | 197.43 | 771.61 | 235.26 | 1231.17 |
| FCDAE-CNN-LSTM | 209.44 | 1466.03 | 205.07 | 2338.9 |
| SVM | 7703 | 316483 | 22541 | 141122 |
| MLP | 561 | 14027 | 480 | 10444 |
| DBN | 418 | 9032 | 442 | 7954 |
| LSTM | 338 | 4450 | 852 | 5550 |
| DCNN | 274 | 10412 | 284 | 12466 |

Fig. 8. Performance comparison of the GBRT I and GBRT III methods with various algorithms from previous literature. The RMSE values for each algorithm are presented on the left, and PHM Score values are presented on the right. (refer to Eq. 2). The best RMSE and PHM Score values are emphasized in red.

GBRT III deliver competitive scores of 1424.28 and 1382.19, respectively, closely aligning with the FCDAE-CNN-LSTM method's score of 1466.03. However, ConvGat significantly outperforms all other methods on FD002 with a score of 771.61.

Overall, GBRT I and GBRT III demonstrate robust predictive accuracy across all four operating scenarios, though their scores are only competitive on FD002 and FD004. Given that these datasets include six operating conditions, GBRT III appears particularly effective at capturing complex degradation patters, though it faces challenges with timely prediction in simpler scenarios.

## VI. RQ3: Performance Metrics

Prognostic methods are application-driven, and their evaluation metrics often vary to suit specific scenarios. As a result, it is challenging to establish a universal evaluation standard [47] [48]. Generally, it's preferred to predict early rather than late in high-stakes environments so that faults can be mitigated as soon as possible, but some systems have economic constraints that make prediction precision more paramount, because while failure is hoping to be avoided, early predictions might result in unnecessary costs. The resulting variance in terminologies and evaluation methods across different studies makes it difficult to establish a fair basis for comparing algorithms even when they are deemed successful within their respective contexts. Without a focused methodology, it is quite difficult to objectively compare the performance of algorithms.

The predictive capability of a prognostic solution is critical, as decision-makers rely on these predictions to inform maintenance strategies. To evaluate solutions, researchers define PPIs to measure the importance of characteristics like prediction accuracy, uncertainty, and precision [41]. However, these PPIs are often aggregated into opaque metrics, which obscure how specific factors influence the overall score.

In the context of turbofan engine prognostics, timeliness of prediction is particularly important. Late predictions may lead to system failures, while overly conservative early predictions can impose unnecessary operational costs. The PHM Score employs an asymmetric scoring function, where penalties for prediction errors grow exponentially and late predictions are penalized more heavily than early ones. While this metric captures the preference for early predictions, it has limitations:

**Transparency.** The PHM Score integrates accuracy and timeliness of prediction into a single score, making it unclear how these factors interact. There is no way to distinguish between a small-magnitude late prediction and a large-magnitude early prediction, obscuring the underlying performance characteristics.

**Standardization.** PHM Score aggregates prediction errors across trajectories without normalization. This means datasets with more trajectories, such as FD002 and FD004, yield unavoidably larger scores than those with fewer trajectories, like FD001 and FD003. Consequently, higher scores may inaccurately imply inferior performance.

**Ambiguity.** Beyond ranking solutions by their score, the metric provides little insight into why one method outperforms another. This lack of detail does not support efforts to improve or tailor prognostic algorithms for specific applications.

The limitations of this metric emphasize the need to develop of a domain-reflective metric that is both transparent and standardized.

### A. Margin-Adjusted Reliability Score

To address these challenges, we designed the Margin-Adjusted Reliability Score (**MARS**), a novel performance metric tailored to the turbofan engine prognostics problem. MARS evaluates the reliability of a RUL predictor by quantifying how well the algorithm performs within a specified margin of error, producing a clear and interpretable measure of performance.

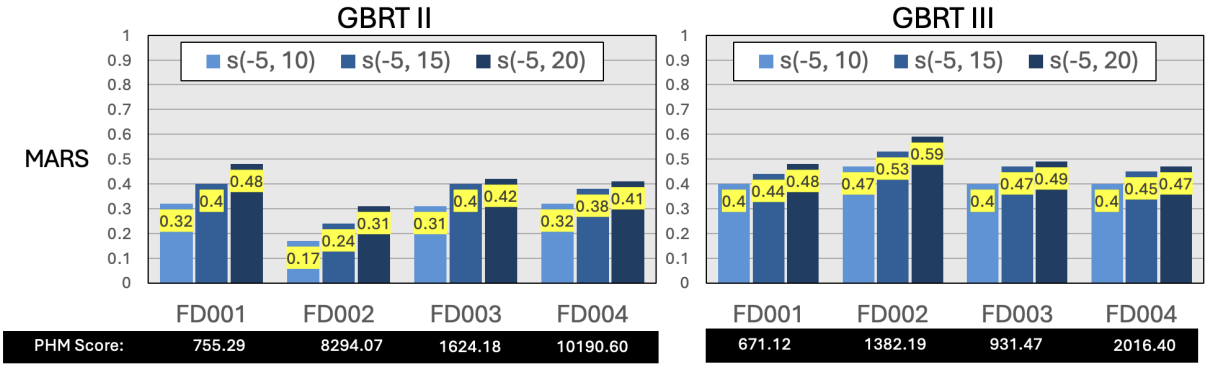



Fig. 9. Comparison of MARS values of GBRT II and GBRT III across sub-datasets FD001 through FD004 using margins (-5, 10), (-5, 15), and (-5, 20). The corresponding score values on each sub-dataset are shown on the bottom.

MARS is defined mathematically in Eq. 4:

$$MARS: s(b_1, b_2) = \frac{1}{n}\sum_{i=1}^{n} I(b_1 \leq d_i \leq b_2)$$
$$d_i = \text{predicted RUL - true RUL for the } i^{th} \text{ trajectory} \quad (4)$$
$$b_1 = \text{lower bound}$$
$$b_2 = \text{upper bound}$$

The indicator function $I(b_1 \leq d_i \leq b_2)$ evaluates to 1 when the prediction error for the trajectory lies within the defined margin $(b_1, b_2)$ and 0 otherwise. MARS scores range from 0 to 1, with values closer to 1 indicating higher reliability and values closer to 0 indicating lower reliability.

MARS explicitly accounts for a margin of maintenance anticipation and evaluates algorithm performance within that margin. This approach provides a standardized measure that penalizes late predictions while offering flexibility in how early predictions are assessed. For example, a margin might allow only a level of prediction uncertainty such that the true RUL to should fall between five below or ten above the predicted RUL ($b_1 = -5$, $b_2 = 10$). In such a case, the necessary actions for fault mitigation can be deferred until the engine's predicted RUL falls to 15, but it must occur before the RUL drops to 5 to ensure safety. With adjustable boundaries of error uncertainty, MARS can balance the trade-off between early and late predictions to reflect the specific requirements.

**Advantages of MARS**: MARS extends the principles of the PHM Score by addressing the following shortcomings:

- **Separation of Prediction Accuracy and Timeliness:** MARS explicitly quantifies how accuracy and timeliness interact to influence the overall score.
- **Standardized Comparisons:** MARS ensures consistent evaluation across any number of trajectories. This standardization allows for objective comparisons, making it easier to benchmark algorithms.
- **Greater Transparency:** MARS clearly illustrates why a particular solution performs well or poorly, offering greater insights into how algorithms can improve.

Figure 9 illustrates MARS results for GBRT II and GBRT III, evaluated with the margin settings of ($b_1 = -5$, $b_2 = 10$), ($b_1 = -5$, $b_2 = 15$), and ($b_1 = -5$, $b_2 = 20$). We can see how comparing MARS values alongside the PHM Score increases our understanding of the performance components.

## VII. Observations and Discussions

In this section, we present and summarize our answers to **RQ1**, **RQ2**, and **RQ3**, discuss the key observations, and give explanations for our experiment results and observed trends.

### A. Observations and Discussions for RQ1

Here, we will present several observations from the outcomes of our investigation into RQ1, and provide further explanation for these results.

**GBRT II exhibits the weakest accuracy and PHM Score across all sub-datasets.** PCA, the linear dimensionality reduction technique applied to the data that GBRT II trained on, is effective at preserving feature variance when feature relationships are predominantly linear. GBRT II's comparable performance on FD001 and FD003, the sub-datasets with the simpler operating scenarios, suggests that the relationships between sensor readings are linear. However, the poor performance on FD002 and FD004 implies that PCA cannot sufficiently preserve the non-linear interactions between sensor readings during complex operating scenarios.

**GBRT I and GBRT III achieve similar accuracy across the sub-datasets.** GBRT I, trained on the original set of features, and GBRT III, trained on polynomial feature mappings, demonstrate comparable accuracy. However, the PHM Score indicates that GBRT III performs better at predicting faults in advance. This suggests that the non-linear interactions captured by the polynomial feature mappings are particularly important for early fault forecasting. While GBRT I achieves a higher PHM score on FD001, the simplest operating scenario, this likely reflects its ability to accurately predict faults that are imminent or obvious.

Our investigation into RQ1 demonstrates the importance of of aligning feature engineering techniques with the underlying complexity of the scenario under study. This suggests that evaluating methods not just on their overall performance but also on their behavior across different conditions is important to examine.

### B. Observations and Discussions for RQ2

Here, we present and explain our observations of the RQ2 outcomes.

**Most methods under study exhibit disparities in accuracy between simple operating conditions (FD001 and FD003) and complex operating conditions (FD002 and FD004), but proposed methods exhibit a much lower level of this variance.** This consistency in the GBRT models' performance is likely due to the inherent strengths of Gradient Boosting Tree algorithms. They excel at identifying feature importance and capturing non-linear relationships between features, which are critical to discern in turbofan engine sensor data. This characteristic implies that GBRT models are a robust solution to make accurate predictions about engine health across a variety of operating conditions.

**While the GBRT models achieve competitive accuracy across all sub-datasets, competitive PHM Scores on FD002 and FD004, they yield relatively poor PHM Scores on FD001 and FD003.** This phenomenon can be attributed to the architecture of the GBRT algorithm. GBRT models iteratively optimize their predictions by correcting errors made in previous iterations, enabling them to capture non-linear relationships and complex interactions between features. This makes them particularly well-suited for learning more complex patterns, such as those in FD002 and FD004, where precise fault detection requires understanding subtle feature relationships and non-linear patterns. On simpler datasets like FD001 and FD003 where feature interactions are less complex, GBRT models' tendency to fit global patterns for maximum accuracy might lead to a more frequent overestimation of the RUL. To alleviate this issue, training the proposed GBRT methods with an asymmetric loss function might encourage less frequent RUL overestimation, though it's generally not advisable to artificially skew the predictions.

The outcome of RQ2 contextualizes the robustness and adaptability of the proposed model, particularly when it comes to prediction within highly complex scenarios. Compared to other methods, the proposed method demonstrates a combination of strong performance, better computational efficiency, and prediction transparency, making them a highly practical and effective choice for real-world applications.

### C. Observations and Discussions for RQ3

**Even on the narrowest margin, GBRT III consistently achieves a MARS value of 0.4 or higher, with its best performance on FD002. However, when measured by the PHM Score, its performance on FD001 and FD003 appears superior to its performance on FD002.** This discrepancy can be attributed to the higher number of trajectories in FD002 compared to FD001 and FD003 (see Table I). The PHM Score aggregates errors across all trajectories, meaning sub-datasets with more trajectories, like FD002, inherently produce larger aggregated error values, which can misleadingly suggest worse performance. In contrast, MARS normalizes results, enabling fair comparisons across sub-datasets regardless of their size. This highlights the importance of employing a normalized metric like MARS, which provides a clearer and more accurate assessment of performance while accounting for the structural differences between sub-datasets.

**MARS indicates that GBRT II's weakest performance occurs on FD002, while the PHM Score suggests its weakest performance is on FD004.** Knowing that FD004 has fewer trajectories than FD002 (Table I) allows us to rule out trajectory number as the cause of this difference. Instead, it suggests that GBRT II exhibits more overestimation of RUL on FD004, which the PHM Score penalizes more heavily. This observation emphasizes the value of MARS in distinguishing performance characteristics and complements the PHM Score by offering additional interpretive context.

The outcomes of RQ3 underscore the complementary power of using both MARS and the PHM Score for evaluating prognostic models. While the PHM Score captures aggregated performance and penalizes late predictions, MARS normalizes results across datasets and provides a nuanced view of accuracy and timeliness within a specified margin. Together, these metrics allow for a deeper understanding of model performance, revealing insights that would otherwise remain obscured if only one metric were used.

Using MARS, researchers can make more informed, data-driven decisions about model selection and refinement.

In the future, it will be beneficial to use this framework to investigate a fully domain-reflective metric. Such a metric could provide a unified, transparent evaluation framework for assessing and improving prognostic solutions.

## VIII. Concluding Remarks

In this study, we propose GBRT-based solutions for forecasting turbofan engine RUL and introduce the MARS evaluation metric to enhance the understanding of turbofan engine fault prognostics systems. The architecture of our proposed method is particularly well-suited for this application due to its exceptional predictive power and algorithmic transparency, which is an essential consideration for high-stakes real-world environments like aviation systems. By analyzing the effects of different dimensionalities of sensor data, we explored how these approaches influence predictive performance across diverse engine operating scenarios. We also show that our proposed method has comparable predictive performance to complex deep learning-based approaches. Additionally, our assessment of our methods using the MARS metric revealed insights that current evaluation metrics fail to capture, showing that MARS offers a more nuanced and industry-relevant perspective for evaluating turbofan engine RUL predictive methods. This work moves towards more effective and reliable predictive maintenance frameworks in aviation. The datasets, tools, and algorithms from this work will be open-sourced to support community research.

## Acknowledgement

This work is supported in part by the National Science Foundation (NSF) grant CNS-2348151 and Commonwealth Cyber Initiative (CCI) grant HC-3Q24-048.

## REFERENCES


[1] D. Vertesy, *The Global Commercial Aviation Industry*. Routledge, 2016, ch. The contours of the global commercial aircraft manufacturing industry.

[2] F. A. Administration, "Air traffic by the numbers," https://www.faa.gov/air_traffic/by_the_numbers, 2024.

[3] M. N. Andrew Potter, Hamad Al-Kaabi, *The Global Commercial Aviation Industry*. Routledge, 2016, ch. Aircraft maintenance, repair, and overhaul.

[4] R. Davis, "Long before jeju air crash, south korea rose to be a model of safety (online)," https://www.nytimes.com/2025/01/06/business/jejuair-crash-south-korea-safety.html, 2025-01-06.

[5] S. E. Daniel Todd, *The Global Commercial Aviation Industry*. Routledge, 2016, ch. Engines.

[6] N. Hall, "Turbojet engine," https://www.grc.nasa.gov/www/k-12/airplane/aturbj.html, 2021.

[7] ——, "Turboprop engine," https://www.grc.nasa.gov/www/k-12/airplane/aturbp.html, 2021.

[8] ——, "Turbofan engine," https://www.grc.nasa.gov/www/k-12/airplane/aturbf.html, 2015.

[9] D. K. Frederick, J. A. DeCastro, and J. S. Litt, "User's guide for the commercial modular aero-propulsion system simulation (c-mapss)," Tech. Rep., 2007.

[10] J. T. Bernardo, "Cognitive and functional frameworks for hard/soft fusion for the condition monitoring of aircraft," in *2015 18th International Conference on Information Fusion (Fusion)*. IEEE, 2015.

[11] S. Lu, X. Yuan, and W. Shi, "Edge compression: An integrated framework for compressive imaging processing on CAVs," in *2020 IEEE/ACM Symposium on Edge Computing (SEC)*. IEEE, 2020, pp. 125–138.

[12] Y. Luo, Y. Yao, J. Chen, S. Lu, and W. Shi, "An efficient data transmission framework for connected vehicles," in *2024 IEEE/ACM Symposium on Edge Computing (SEC)*. IEEE, 2024, pp. 306–320.

[13] B. Huang, Y. Di, C. Jin, and J. Lee, "Review of data-driven prognostics and health management techniques: Lessions learned from phm data challenge competitions," 2017.

[14] S. Lu, Y. Yao, and W. Shi, "CLONE: Collaborative learning on the edges," *IEEE Internet of Things Journal*, vol. 8, no. 13, pp. 10 222–10 236, 2020.

[15] R. Walthall and R. Rajamani, *The Role of PHM at Commercial Airlines*. John Wiley & Sons, Ltd, 2018, ch. 18, pp. 503–534. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1002/9781119515326.ch18

[16] J. Dalzochio, R. Kunst, J. L. V. Barbosa, P. C. d. S. Neto, E. Pignaton, C. S. ten Caten, and A. d. L. T. da Penha, "Predictive maintenance in the military domain: A systematic review of the literature," *ACM Comput. Surv.*, 2023.

[17] J. Chen and S. Lu, "An advanced driving agent with the multimodal large language model for autonomous vehicles," in *2024 IEEE International Conference on Mobility, Operations, Services and Technologies (MOST)*. IEEE, 2024, pp. 1–11.

[18] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation," in *2008 International Conference on Prognostics and Health Management*, 2008, pp. 1–9.

[19] S. Lu, B. Luo, T. Patel, Y. Yao, D. Tiwari, and W. Shi, "Making disk failure predictions SMARTer!" in *18th USENIX Conference on File and Storage Technologies (FAST 20)*, 2020, pp. 151–167.

[20] X. Chen and M. Zeng, "Convolution-graph attention network with sensor embeddings for remaining useful life prediction of turbofan engines," *IEEE Sensors Journal*, vol. 23, no. 14, pp. 15 786–15 794, 2023.

[21] Y. Wang and Y. Wang, "A denoising semi-supervised deep learning model for remaining useful life prediction of turbofan engine degradation," *Applied Intelligence*, vol. 53, no. 19, pp. 22 682–22 699, 2023.

[22] C. Zhang, L. Pin, A. Qin, and K. Tan, "Multiobjective deep belief networks ensemble for remaining useful life estimation in prognostics," *IEEE Transactions on Neural Networks and Learning Systems*, vol. PP, pp. 1–13, 07 2016.

[23] C. Chen, J. Shi, N. Lu, Z. H. Zhu, and B. Jiang, "Data-driven predictive maintenance strategy considering the uncertainty in remaining useful life prediction," *Neurocomputing*, vol. 494, 04 2022.

[24] Q. Zhang, L. Yang, W. Guo, J. Qiang, C. Peng, Q. Li, and Z. Deng, "A deep learning method for lithium-ion battery remaining useful life prediction based on sparse segment data via cloud computing system," *Energy*, vol. 241, p. 122716, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0360544221029650

[25] Z. Xu and J. H. Saleh, "Machine learning for reliability engineering and safety applications: Review of current status and future opportunities," *Reliability Engineering System Safety*, vol. 211, p. 107530, 2021. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0951832021000892

[26] D. An, "A practical prognostics method based on stepwise linear approximation of a nonlinear degradation model," *Applied Sciences*, vol. 15, no. 1, 2025. [Online]. Available: https://www.mdpi.com/2076-3417/15/1/266

[27] J. Philip *et al.*, "Cnn-lstm hybrid deep learning model for remaining useful life estimation," *arXiv preprint arXiv:2412.15998*, 2024.

[28] Y. Luo, D. Xu, G. Zhou, Y. Sun, and S. Lu, "Impact of raindrops on camera-based detection in software-defined vehicles," in *2024 IEEE International Conference on Mobility, Operations, Services and Technologies (MOST)*. IEEE, 2024, pp. 193–205.

[29] J. Hong, Z. Wang, W. Chen, and Y. Yao, "Synchronous multi-parameter prediction of battery systems on electric vehicles using long short-term memory networks," *Applied Energy*, vol. 254, p. 113648, 2019.

[30] B. Najdi, M. Benbrahim, and M. N. Kabbaj, "Adaptive res-lstm attention-based remaining useful lifetime prognosis of rolling bearings," *International Journal of Prognostics and Health Management*, vol. 16, no. 1, 2025.

[31] W. Shuai, L. Yiting, Z. Shoubin, C. Lifei, and M. Pecht, "Remaining useful life prediction of lithium-ion batteries using a novel particle flow filter framework with grey model," 2024.

[32] E. Ramasso and A. Saxena, "Review and Analysis of Algorithmic Approaches Developed for Prognostics on CMAPSS Dataset," in *Annual Conference of the Prognostics and Health Management Society 2014.*, Fort Worth, TX, USA., United States, Sep. 2014. [Online]. Available: https://hal.science/hal-01145003

[33] X. Cao, K. Peng, and R. Jiao, "Degradation modeling and remaining life prediction for a multi-component system under triple uncertainties," *Computers & Industrial Engineering*, p. 110432, 2024.

[34] S. Onofri, A. Marchioni, G. Setti, M. Mangia, and R. Rovatti, "Multi-class similarity-based approach for remaining useful life estimation," in *2024 IEEE International Instrumentation and Measurement Technology Conference (I2MTC)*, 2024, pp. 01–06.

[35] Z. Cen, S. Hu, Y. Hou, Z. Chen, and Y. Ke, "Remaining useful life prediction of machinery based on improved sample convolution and interaction network," *Engineering Applications of Artificial Intelligence*, vol. 135, p. 108813, 2024. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0952197624009710

[36] P. Khumprom, A. Davila-Frias, D. Grewell, and D. Buakum, "A hybrid evolutionary cnn-lstm model for prognostics of c-mapss aircraft dataset," in *2023 Annual Reliability and Maintainability Symposium (RAMS)*, 2023, pp. 1–8.

[37] M. Akbari Pour and M. S. Karimi, "Temporal convolutional and fusional transformer model with bi-lstm encoder-decoder for multi-time-window remaining useful life prediction," *Available at SSRN 5059977*.

[38] R. Wang, Y. Zhang, C. Hu, Z. Yang, H. Li, F. Liu, L. Li, and J. Guo, "A parallel prognostic method integrating uncertainty quantification for probabilistic remaining useful life prediction of aero-engine," *Processes*, vol. 12, no. 12, 2024. [Online]. Available: https://www.mdpi.com/2227-9717/12/12/2925

[39] C. He, L. Chen, N. Sun, P. Chen, X. Xu, and S. Lu, "Redundancy modification and potential feature reactivation network for predicting the remaining useful life of machines," *IEEE Sensors Journal*, 2024.

[40] A. Behbahani, S. Adibhatla, and C. Rauche, "Integrated model-based controls and phm for improving turbine engine performance, reliability, and cost," *45th AIAA/ASME/SAE/ASEE Joint Propulsion Conference and Exhibit*, p. 28, 09 2009.

[41] E. Zio, "Prognostics and health management (phm): Where are we and where do we (need to) go in theory and practice," *Reliability Engineering System Safety*, vol. 218, p. 108119, 2022. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0951832021006153

[42] C. Aerospace, "Engine system sensors data sheet."

[43] F. Heimes, "Recurrent neural networks for remaining useful life estimation," 11 2008, pp. 1 – 6.

[44] T. Wang, J. Yu, D. Siegel, and J. Lee, "A similarity-based prognostics approach for remaining useful life estimation of engineered systems," 11 2008, pp. 1 – 6.

[45] Z. Huang, H. Zheng, C. Li, and C. Che, "Application of machine learning-based k-means clustering for financial fraud detection," *Academic Journal of Science and Technology*, vol. 10, no. 1, pp. 33–39, 2024.

[46] R. Munagala, "Gradient boost for regression explained," 2021, available: https://www.numpyninja.com/post/gradient-boost-for-regression-explained.

[47] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha, and M. Schwabacher, "Metrics for evaluating performance of prognostic techniques," in *2008 International Conference on Prognostics and Health Management*, 2008, pp. 1–17.

[48] A. Saxena, J. Celaya, B. Saha, S. Saha, and K. Goebel, "Metrics for offline evaluation of prognostic performance," *International Journal of Prognostics and health management*, vol. 1, no. 1, pp. 4–23, 2010.

[49] G. Sateesh Babu, P. Zhao, and X.-L. Li, "Deep convolutional neural network based regression approach for estimation of remaining useful life," in *Database Systems for Advanced Applications*, S. B. Navathe, W. Wu, S. Shekhar, X. Du, X. S. Wang, and H. Xiong, Eds. Cham: Springer International Publishing, 2016, pp. 214–228.

[50] H. Wang, D. Li, D. Li, C. Liu, X. Yang, and G. Zhu, "Remaining useful life prediction of aircraft turbofan engine based on random forest feature selection and multi-layer perceptron," *Applied Sciences*, vol. 13, no. 12, p. 7186, 2023.

[51] L. Xingqiu, H. Jiang, Y. Liu, T. Wang, and Z. Li, "An integrated deep multiscale feature fusion network for aeroengine remaining useful life prediction with multisensor data," *Knowledge-Based Systems*, vol. 235, p. 107652, 10 2021.

[52] X. Li, Q. Ding, and J. Q. Sun, "Remaining useful life estimation in prognostics using deep convolution neural networks," *Reliability Engineering System Safety*, vol. 172, 12 2017.