Improving Predictive Maintenance with the Health-Aware Transformer

Islam M. Momtaz A. Sadek[1], Eric Postma[2], Rogier Brussee[3], and Juan Sebastian Olier[2]

[1] Jheronimus Academy of Data Science (JADS), Tilburg University and Eindhoven University of Technology, The Netherlands
I.Momtaz@tilburguniversity.edu
[2] Tilburg University, The Netherlands
E.O.Postma@tilburguniversity.edu
[3] Jheronimus Academy of Data Science (JADS), The Netherlands
R.Brussee@jads.nl
[4] Tilburg University, The Netherlands
J.S.Olier@tilburguniversity.edu

Abstract. Accurate prediction of the remaining useful life (RUL) of industrial machinery is central to predictive maintenance. The best RUL prediction accuracy reported in the literature is an RMSE of 11.27 on the NASA C-MAPSS benchmark, achieved by models such as the GCU-Transformer. However, these models act as black boxes with limited interpretability, which limits their trust in safety-critical applications.

This study presents the Health-Aware Transformer (HAT), an extension of the Gated Convolutional Unit–Transformer that improves prediction accuracy while introducing transparency. HAT integrates a statistical framework based on the Mahalanobis Distance (MD), which quantifies deviations from a multivariate Gaussian baseline and serves as a clear health degradation indicator. The MD guides the model's attention toward cycles that significantly deviate from a healthy baseline, linking the prediction to observable physical degradation.

The Health-Aware Transformer achieves an RMSE of 10.95 and a safety rate of 38 safe predictions out of 100, outperforming existing models including BiLSTM-Attention (13.21), CTVAE (12.41), and the original GCU-Transformer (11.27). By embedding MD into the attention mechanism, HAT enhances predictive accuracy while slightly reducing safety, reflecting the trade-off between precision and conservative early warnings.

As a secondary analysis, without the Gated Convolutional Unit–Transformer, the statistical ensemble of regressors based on MD trajectories achieves an RMSE of 15.51 and a safety rate of 53 engines. This interpretable model is well suited for safety-critical contexts requiring conservative predictions.

Overall, the study quantifies the gap between transparent statistical models and complex deep learning approaches: accuracy improves from 15.51 (statistical ensemble) to 11.27 (GCU-Transformer) and further to 10.95 (HAT), showing concretely how predictive accuracy increases with model complexity.

Keywords: Artificial Intelligence · Predictive Maintenance · Remaining Useful Life (RUL) · Prognostics and Health Management (PHM)· Mahalanobis Distance · Health-Aware Transformer (HAT) · Explainable AI (XAI) · NASA C-MAPSS Dataset

1 Introduction

Industrial systems face costly risks when mechanical components fail unexpectedly. Breakdowns lead to downtime, repair costs, and safety hazards. Predicting the time of failure is therefore central to maintaining reliability and efficiency. Predictive maintenance addresses this challenge by forecasting failures before they occur. A key task is predicting the Remaining Useful Life (RUL), the time until a system can no longer operate reliably. Accurate RUL prediction enables early warning and better maintenance planning, reducing operational risk. Research on RUL prediction has advanced from traditional machine learning to deep learning. Early models such as Support Vector Machines[1], Multi-Layer Perceptrons[17], and Extreme Learning Machines[29] relied on manually engineered statistical and frequency features. These approaches were sensitive to feature choice and struggled with nonlinear, high-dimensional sensor data.

Deep learning removed the dependency on manual features. Recurrent Neural Networks (RNNs)[9], Convolutional Neural Networks (CNNs)[18][28], and Long Short-Term Memory networks [25] captured nonlinearities and temporal dependencies, improving predictive accuracy. More recently, Transformer and Informer architectures were successfully applied to RUL prediction [31,24]. The improved RUL prediction accuracy comes at the cost of limited explainability, as is the case with most deep learning approaches.

Addressing this limitation, standard statistical approaches may provide an interpretable foundation, while sacrificing accuracy. During the baseline period of engine operation, sensor signals can be modeled as a multivariate Gaussian distribution. The Mahalanobis Distance (MD) then serves as a natural index of deviations in terms of standard deviations from the mean. As such, MD constitutes an engine degradation metric. The interpretability in our approach comes from this MD measure, as it produces a direct, physically meaningful health score for each engine cycle that can be inspected independently of the neural network. This allows every prediction to be linked to an observable deviation from the healthy baseline rather than to hidden network activations.

In this paper, we explore whether combining the MD metric with Transformers can improve RUL prediction accuracy while maintaining interpretability. The combined method is called the Health-Aware Transformer (HAT) and endows a Transformer with health-aware features derived from the MD metric. Through attention, HAT prioritizes cycles that significantly deviate from the healthy baseline, creating a direct correspondence between attention weights and degradation levels.

The remainder of this paper is organized as follows. Section 2 reviews related work. Section 3 presents the Health-Aware Transformer. Section 4 describes the experimental method and Section 5 presents the results. Finally, Section 6 discusses our work and draws conclusions.

2 Related Work

This section provides an overview of recent studies on the prediction of RUL, with a particular focus on deep learning architectures and statistical methods that inform the design and capabilities of our Health-Aware Transformer (HAT). For a broader review of machine learning approaches to RUL prediction, see [20]. We highlight how existing models advance predictive accuracy and interpretability, and identify the gaps that HAT addresses by integrating statistically-grounded health indicators into a Transformer architecture.

Recent studies in RUL predictions combine Transformer-based methods with specialized enhancements to capture complex degradation dynamics from multivariate time-series data. While these advanced deep learning approaches, exemplified by the TI-former [24], ATTN+LSTM [3], and CTVAE [21], have significantly improved RUL prediction accuracy by capturing complex temporal dependencies and non-linearities, they often achieve this at the cost of limited explainability, a common challenge in most deep learning models. Our Health-Aware Transformer (HAT) builds upon these advancements but seeks to enhance interpretability by explicitly incorporating a health indicator.

Li et al. [3] introduced an attention-enhanced LSTM framework for RUL prediction on turbofan engines. Their model utilizes stacked LSTM layers to learn temporal dynamics from sensor sequences, while an attention mechanism identifies crucial time steps that contribute most to the final prediction. This approach allows the model to weigh informative degradation regions more heavily than early-cycle signals, thereby improving accuracy. The authors evaluated their method on the FD001 subset of the NASA C-MAPSS dataset, achieving competitive RMSE performance and retaining interpretability through attention visualizations. This work highlights the growing trend of combining recurrent architectures with adaptive focus mechanisms to improve both performance and interpretability in RUL prediction. This demonstrates the value of attention mechanisms in RUL prediction, a principle that our HAT model extends by explicitly guiding attention with statistically-derived health features.

Mo et al. [15] proposed a framework for RUL prediction that combines a Transformer encoder with a Gated Convolutional Unit (GCU). This combination simultaneously captures long-term dependencies and local contextual information in time-series data. Unlike traditional CNN- or RNN-based approaches, which may suffer from limited receptive fields or sequential processing constraints, their model leverages the Transformer's self-attention mechanism to access global dependencies across all time steps without performance bottlenecks. To address the insensitivity of Transformer outputs to neighboring time steps, a GCU module was introduced, incorporating reset and update gates to re-emphasize local feature importance. Experiments on the C-MAPSS datasets demonstrated that the GCU-Transformer achieved superior performance, obtaining the lowest Root Mean Square Error (RMSE) on FD001 (11.27) and FD003 (11.42), surpassing previous methods like CNN-FNN (RMSE 12.61) and LSTM-FNN (RMSE 16.14). Averaged over all datasets, the model reached an RMSE of 17.59, ranking second overall while maintaining consistent gains under simple and complex operating conditions. An ablation study further confirmed the effectiveness of both architectural enhancements: removing the GCU module increased the RMSE by 2.15 on FD002, and eliminating the output Sigmoid activation led to an additional 0.61 RMSE degradation. This work is particularly relevant to our study because the GCU-Transformer serves as the foundational baseline architecture that our Health-Aware Transformer (HAT) extends and directly compares against. Its success, demonstrated by achieving superior performance with an RMSE of 11.27 on FD001, established a strong benchmark for Transformer-based RUL prediction. The model's ability to leverage Transformer's self-attention for global dependencies while using GCU for local feature importance provides a powerful base, highlighting the potential for further improvements by integrating explicit health awareness into its attention mechanism.

Hangjun Wu et al. [24] proposed TI-former, an end-to-end RUL prediction model that integrates Transformer and Informer components. While Transformers capture global dependencies through full self-attention, they scale quadratically with sequence length ($O(L^2)$). Informer mitigates this by using ProbSparse Attention, which focuses on dominant queries and reduces complexity to $O(L \log L)$ while retaining essential dependencies. TI-former combines a Transformer encoder for global feature extraction with an Informer decoder for efficient long-range temporal modeling. It also introduces a learnable query vector that guides decoding attention toward relevant parts of the sequence, enhancing robustness under noise and high dimensionality. The model was evaluated on the XJTU-SY Bearing dataset and the ETDataset of electrical equipment degradation. On XJTU-SY (35Hz–12kN), it achieved an RMSE of 0.0941, outperforming CNN-Informer (0.1163). On ETDataset, it further improved RMSE and computational efficiency, demonstrating its generalization across domains. The TI-former's focus on efficient attention mechanisms and enhanced robustness is a critical advancement, and HAT similarly aims for more effective attention by explicitly guiding it with health-aware features.

Wang et al. [21] proposed the Convolutional Transformer Variational Autoencoder (CTVAE) for RUL prediction. This model integrates 1D CNNs for local feature extraction with Transformer encoders for global temporal dependencies. Unlike standard Transformers, its embedding layer encodes each sensor's full signal as a feature vector, which enhances inter-sensor attention. A Variational Autoencoder (VAE) projects learned features into a two-dimensional latent space, allowing engines to form smooth degradation trajectories from healthy to failure clusters. This provides interpretable health progression and supports anomaly detection. A lightweight MLP then regresses these latent features to scalar RUL values. On C-MAPSS, CTVAE achieved a competitive RMSE of 12.41 on FD001 (close to GCU-Transformer at 11.27). It also outperformed CNN, LSTM, and Transformer baselines on FD002 (14.24 vs. 22.81–24.49) and FD004 (15.70 vs. 24.86–28.17), demonstrating stronger generalization across complex fault modes. While CTVAE offers interpretability through learned latent space trajectories, HAT pursues a different, more direct form of health awareness by integrating a statistically-grounded Mahalanobis Distance metric into the attention mechanism, aiming for transparent health indicators that directly guide predictions, distinguishing our approach from these existing methods.

Hybrid architectures have expanded RUL prediction beyond single paradigms. Fan et al. [6] combined a Bidirectional LSTM autoencoder with a Transformer encoder under a self-supervised denoising framework, which improved robustness to noise. A Transformer-GRU network was also proposed for aero-engine prognostics, utilizing GRU units to stabilize Transformer outputs across degradation sequences [12]. Other works explore LSTM and Transformer based methods for RUL prediction, considering challenges such as censored data [16]. Our work aligns with this trend of hybrid models, but specifically focuses on integrating statistical methods with Transformer architectures to create a 'health-aware' model, thereby addressing the acknowledged trade-off between accuracy and interpretability.

Variational methods have been applied for interpretability in RUL prediction. Building on foundational work in variational autoencoders [10], Costa and Sánchez [2] projected health trajectories into a low-dimensional latent space using a VAE. Similarly, Xiang et al. [26] integrated temporal–channel fusion with variational encoding to disentangle degradation dynamics. Both approaches demonstrated interpretable latent trajectories that aligned with degradation progression. Furthermore, conditional variational transformers have been developed for bearing RUL prediction, focusing on selecting important features correlated with RUL [22]. These methods clearly demonstrate a trend toward enhancing the interpretability of RUL models through learned representations. However, they achieve interpretability primarily through abstract latent spaces. Our work with HAT seeks to provide a complementary and more explicit form of interpretability by directly integrating a well-understood, statistically-grounded degradation metric (Mahalanobis Distance) into the model's attention mechanism, offering a transparent health indicator that directly guides predictions.

Mahalanobis Distance (MD) has been widely applied in prognostics and health monitoring as both a similarity metric and a health indicator. Early applications include the Mahalanobis-Taguchi System approach for RUL prediction [8]. In similarity-based RUL prediction, MD provides a multivariate measure for matching feature vectors [32]. Extensions such as weighted MD (WMD) have been utilized for ball screw systems to emphasize defect-relevant features [13]. In bearing prognostics, MD has served as a sensitive health index when combined with GRU-based models [30]. The extensive application of MD in various prognostic tasks, including its extensions like WMD [13] and its combination with GRU-based models [30], validates its effectiveness as a robust, interpretable health indicator capable of quantifying multivariate deviations from a healthy baseline. This inherent capability to account for inter-sensor correlations makes MD an ideal candidate for forming the explicit health-aware features in our HAT model.

Despite its proven utility, the direct integration of MD into deep learning models for RUL prediction remains limited. This identified gap—the employment of MD mostly as a separate thresholding mechanism or a standalone health indicator rather than deeply integrated—is precisely what our Health-Aware Transformer aims to address. While other domains, such as NLP and fault diagnosis, have successfully demonstrated the benefits of coupling MD with modern sequence models like Transformer encoders [11,?], its potential for deeply guiding attention in RUL prediction to simultaneously improve accuracy and offer transparent health insights remains largely unexplored in this specific domain, thus motivating our HAT model.

MD has also been employed for multivariate outlier detection. Dashdondov and Kim [4] applied an MD-based thresholding approach to remove abnormal samples in hypertension prediction. Their method combined feature selection, MD outlier detection ($p < 0.001$), and classification with Random Forest and XGBoost, improving accuracy, F1-score, and AUC. Although this example comes from the healthcare domain, it reinforces MD's ability to identify significant deviations and highlight informative structure—principles that are directly transferable and crucial to our approach in HAT, where MD signals deviations from a healthy state to guide the Transformer's attention towards critical degradation phases in RUL prediction.

Recent surveys reinforce the potential of integrating statistical and deep learning methods. Wu et al. [23] reviewed deep learning methods for RUL, emphasizing Transformer- and VAE-based models. Similarly, Zhou et al. [32] summarized health-indicator-driven prediction strategies for rotating machinery. Crucially, both surveys identify the persistent trade-off between predictive accuracy and interpretability. These findings collectively corroborate the fundamental motivation behind our Health-Aware Transformer (HAT): the need to bridge this gap by integrating statistically-grounded measures like MD into deep learning pipelines for RUL prediction, thereby validating our hybrid approach.

3 The Health-Aware Transformer

In this section, we present the Health-Aware Transformer that consists of two main components: Health-aware features derived from the multi-variate statistics of sensor data and the MD metric (Section 3.1), and the Transformer model in which the health-aware features guide attention weights (Section 3.2).

3.1 Health-aware Features

The health-aware features are defined in terms of the multivariate Gaussian distribution and the joint covariance of the multidimensional sensor time series. The Mahalanobis Distance (MD) is used as a metric to quantify deviations from a healthy baseline [14],[5], [27].

The healthy state of the sensor signals is assumed to correspond to the initial part of the sensor time series. We have empirically verified that the first 25% of the time series provides a good choice (see the Appendix for further details).

The health-aware features are formally defined as follows. For two vectors x and y, the squared MD is defined as:

$$d_M^2(x, y) = (x - y)^\top \Sigma^{-1}(x - y). \tag{1}$$

with vectors $x, y \in \mathbb{R}^d$ and Σ the covariance matrix. In the HAT, x represent the sensor signals and y the baseline mean vector (healthy state) μ, giving the squared health-aware MD:

$$MD = d_M(x_t, \mu) = \sqrt{(x_t - \mu)^\top \Sigma^{-1}(x_t - \mu)}. \tag{2}$$

The health-aware feature MD captures the sensor correlations and feature variances, reflecting the inter-dependencies among sensor signals and providing a direct, interpretable measure of how far each engine state deviates from the healthy baseline.

As an illustration of the validity of the health-aware feature, we plot the MD of the multivariate sensor readings of our dataset. Figure 1 shows the development of MD as a function of cycle number for one engine (Engine 69) that fails at cycle 362. During the initial 25% of the cycles (health state), the MD is relatively stable and increases as degradation progresses towards failure at cycle 362. The illustration suggests that health-aware features provide a reliable estimate of RUL, and that each MD value serves as a transparent health score that can be inspected independently of the neural network.

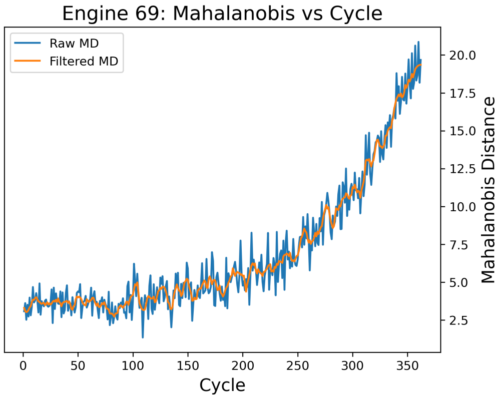


Fig. 1: Mahalanobis Distance (raw and smoothed) for Engine 69.

3.2 Health-aware Attention

We build our Health-Aware Transformer (HAT) on the GCU-Transformer [15]. It has two parts: (1) a Gated Convolutional Unit (GCU) with 128 filters and kernel size 3, which reduces noise and highlights degradation patterns; (2) a Transformer encoder with

1 layer, 4 heads, and 128 hidden dimensions, followed by a regression head for normalized RUL.

HAT adds health-aware features via an MD-based bias in self-attention. This focuses attention on cycles far from the healthy baseline. Let $Q, K \in \mathbb{R}^{T \times d}$ be queries and keys for sequence length T, and $m \in \mathbb{R}^T$ the smoothed MD sequence. Here, attention weights represent the learned importance scores indicating how much the model focuses on different cycles within the input sequence. The attention logits S become:

$$S = \frac{QK^\top}{\sqrt{d}} + \alpha \tanh(Wm)\mathbf{1}^\top, \tag{3}$$

where W projects m per head, tanh squashes to $[-1, 1]$, α scales the bias, and $\mathbf{1}^\top$ broadcasts it. Final attention is:

$$\text{Attention}(Q, K, V) = \text{softmax}\left(\frac{S}{\tau}\right) V, \tag{4}$$

with $\tau = 1.2$ for smoother weights. The MD bias boosts logits for high-MD (degraded) cycles.

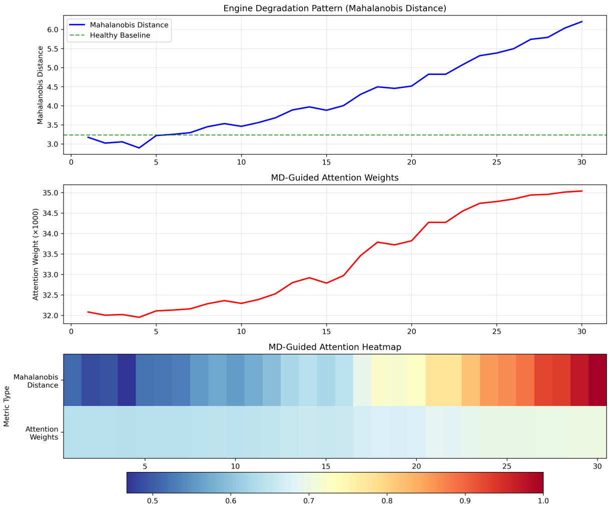


Fig. 2: Conceptual illustration of health-aware attention in HAT. Mahalanobis Distance rises with degradation and biases attention to focus on later cycles.

Figure 2 provides a conceptual illustration of health-aware attention using a simulated degradation pattern to demonstrate the intended mechanism. The top panel shows Mahalanobis Distance (MD, blue) starting near the healthy baseline (green dashed) and rising with degradation. The MD sequence m enters the bias term of self-attention in Equation 3. The middle panel shows per-cycle attention weights (red), i.e., the average attention mass a cycle receives across heads and queries after biasing and temperature softmax. MD is projected Wm, squashed tanh, scaled α, broadcast $\mathbf{1}^\top$, and added to the attention logits as in Equation 3.

The softmax with temperature $\tau = 1.2$ in Equation 4 smooths the distribution and increases focus on degraded cycles. Here, α controls the strength of the MD-based bias, determining how strongly health information influences the attention weights, while τ

scales the attention logits before softmax normalization. Both parameters were empirically selected through validation to balance interpretability and predictive precision. The bottom heatmap compares MD in the top row and the corresponding per-cycle attention weights in the bottom row. The single red line is the per-cycle summary, while the heatmap places these values next to MD for comparison.

4 Experimental Set-up

In this section we provide the details about the implementation and experimental evaluation of the health-aware features and of the complete HAT.

4.1 Dataset

The Health-Aware Transformer (HAT) is evaluated on RUL prediction using the NASA C-MAPSS dataset [19]. This dataset simulates turbofan engine degradation with 21 sensors and 3 operational settings. Table 1 summarizes four subsets (FD001–FD004). FD001, our focus, has 100 training and 100 testing engines, one operating condition, and one fault mode (high-pressure compressor). FD002–FD004 add more conditions or faults.

Table 1: C-MAPSS subsets.

Subset	**FD001**	**FD002**	**FD003**	**FD004**
Training Engines	100	260	100	249
Testing Engines	100	259	100	248
Operating Conditions	1	6	1	6
Fault Modes	HPC	HPC	HPC Fan	HPC Fan

The NASA C-MAPSS FD001 subset is a widely used benchmark in Prognostics and Health Management (PHM) research. It is structured to ensure a clear separation between model development and evaluation, and includes:

1. A training set with full run-to-failure sensor trajectories for 100 engines.
2. A test set with truncated sensor sequences for another 100 engines, where the true Remaining Useful Life (RUL) is unknown.
3. A ground-truth file with the actual RUL values for each test engine, used only for final model evaluation.

This strict division ensures unbiased performance assessment and supports rigorous experimental protocols.

Each engine (train or test) is represented by 21 sensor measurements. Figure 3 shows the histograms of engine 69 sensors, the longest-lived engine in the training set. Narrow spike distributions (e.g., `sensor_1`) indicate zero variance and no informative value. Bell-shaped distributions (e.g., `sensor_2`) suggest stable behavior suitable for health modeling. Based on variance and trend analysis, we retain the following 14 sensors:

`sensor_2`, `sensor_3`, `sensor_4`, `sensor_7`, `sensor_8`, `sensor_9`, `sensor_11`, `sensor_12`, `sensor_13`, `sensor_14`, `sensor_15`, `sensor_17`, `sensor_20`, `sensor_21`.

These sensors exhibit stable variance and clear degradation behavior. Figure 4 shows the correlation structure among them for Engine 69's baseline (first 25% of its cycles), confirming that many of these sensors are jointly informative of engine health.

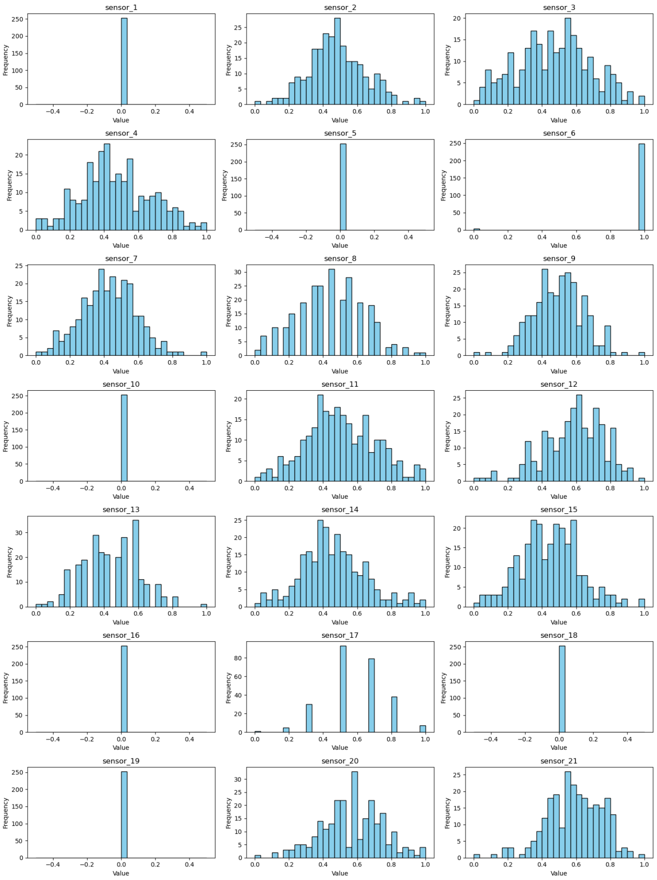


Fig. 3: Histograms for Engine 69. Spikes show zero-variance sensors (e.g., `sensor_1`); bell shapes show Gaussian-like sensors (e.g., `sensor_2`).

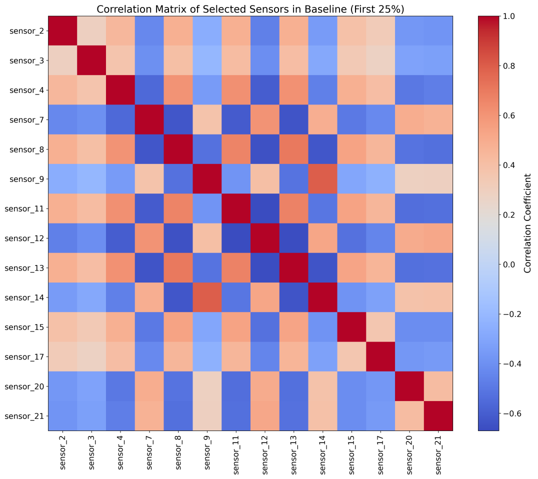


Fig. 4: Heatmap of sensor correlations for Engine 69 in baseline (first 25%), showing relationships among selected sensors with colors: red for high similarity, blue for differences.

4.2 Health-Aware Transformer

The experimental setup of the Health-Aware Transformer (HAT) is described in two parts: the Health-Aware Features (Section 4.3) and the Health-Aware Attention (Section 4.4).

4.3 Health-Aware Features

Building on the Mahalanobis Distance (MD) described in Section 3.1, this approach quantifies deviations from the healthy state of the engines. The baseline distribution is estimated from the first 25% of cycles in the training set (see Appendix). MD values are then computed for each subsequent cycle to track degradation. These MD features are used as input for the second part of HAT. Below, we detail the experimental procedure for evaluating RUL prediction using MD features alone.

Prediction Strategy. The prediction strategy follows a two-model framework for mapping MD to RUL. In the first model, the last-cycle prediction, the final MD value of each test engine is mapped to an Predicted Remaining Useful Life (RUL), denoted as ˆRUL, which is then capped at $\min(125, \hat{RUL})$. In the second model, a threshold-aware moving average, a degradation threshold of $d_M \geq 8$ marks the onset of degradation. This threshold was empirically determined on the training set as the most stable value for balancing early detection against false alarms, with sensitivity analysis indicating that small variations did not significantly alter performance. When an engine does not cross this threshold, predictions default to the last-cycle approach. Once the threshold is crossed, subsequent MD values are converted into RUL mapped values, each capped at $\min(125, \hat{RUL})$. The final RUL is then computed as the moving average of the last five predictions. Both strategies are applied with the two regressors, and their outputs are merged by ensemble weighting (Figure 5).

Regression Models and Tuning. Two regressors were implemented. The first, *Direct Bin Mapping*, groups training pairs (d_M, RUL) into bins of width 0.1. The median RUL in each bin defines an empirical mapping curve. Bins with insufficient samples are discarded, and the degradation threshold $(d_M \geq 8)$ is applied as described above. The second, *Gradient Boosting*, is implemented with 500 estimators, maximum depth of 2, and a learning rate of 0.2. Each regressor was applied under both prediction strategies: last-cycle and threshold-aware moving average.

Ensemble and Weighting Strategy. Final predictions were obtained by weighted averaging of the two regressors. Weights were optimized by constrained grid search under the condition $\sum w_i = 1$. The selected configuration assigned higher weight to Gradient Boosting, with Direct Bin Mapping providing complementary stability after the degradation threshold.

4.4 Health-Aware Attention

Health-Aware Transformer Architecture. The Health-Aware Transformer (HAT) is built upon the GCU-Transformer [15] and processes input data from 14 selected sensors, segmented into overlapping 30-cycle windows. Its architecture incorporates a Gated Convolutional Unit (GCU) with 128 filters and a kernel size of 3, which reduces noise and highlights degradation patterns. This is followed by a single Transformer encoder layer employing 4 attention heads and a hidden size of 128. The unique aspect of HAT is a learned health-aware bias, derived from a smoothed Mahalanobis Distance (MD) trajectory, that modulates cycle-to-cycle attention weights within the Transformer encoder. The MD trajectory itself is smoothed using a Savitzky–Golay filter (window of 7,

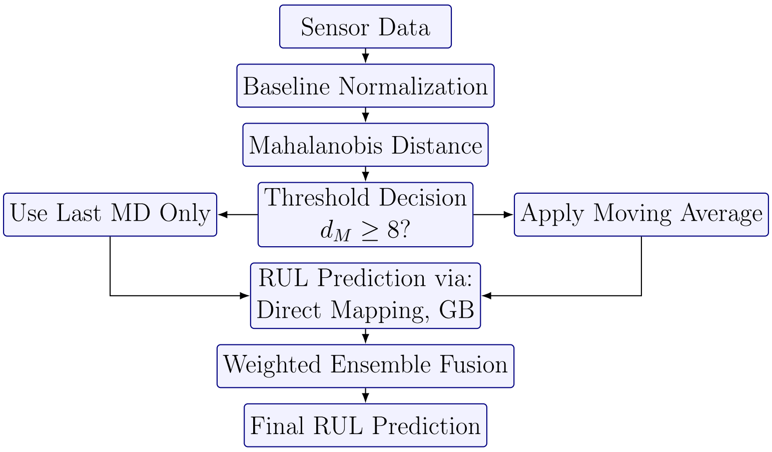


Fig. 5: Pipeline for RUL prediction from Health-Aware Features. Mahalanobis Distance decides between last-cycle mapping and moving-average smoothing; both feed the ensemble.

polynomial order of 3). Finally, a fully connected linear layer maps the encoder output to normalized RUL values, rescaled to a maximum of 125 cycles.

Encoder Depth. A single Transformer encoder layer was found optimal, achieving an RMSE of 10.71 on the FD001 dataset. Increasing to two layers elevated the error, suggesting that the MD-guided bias sufficiently captures long-term degradation information, reducing the need for deeper architectures.

Training Loss. The model is trained using Mean Squared Error (MSE) loss with capped RUL labels (≤ 125). The cap reduces variance in early-cycle predictions, consistent with prior work [19,15].

Optimizer and Scheduler. Optimization is performed with Adam at learning rate 1.42×10^{-4}. A linear learning rate warm-up followed by constant schedule is applied.

Training Protocol. Experiments were conducted using the designated FD001 training and test datasets. The HAT model was trained on the training data file (100 full life engines), with this set further split into 90/10 into training and validation sets. Following the training and validation phase, the model's final performance was evaluated in the separate test data file (100 distinct engines), with predictions for each test engine subsequently compared against the corresponding actual RUL values from the dedicated actual RUL file, using the metrics detailed in the next sub-section 4.5.

Hyperparameters. The fixed hyperparameters across experiments include an input sequence of 30 cycles, one Transformer layer with 4 heads and hidden size 128, a batch size of 32, MSE loss with a 125-cycle RUL cap, a temperature of $\tau = 1.2$ for scaled logits, and an attention bias initialized at 0.03.

4.5 Evaluation Metrics

To evaluate RUL prediction performance, we report the following three metrics:

Root Mean Squared Error (RMSE) quantifies the average squared difference between predicted and true RUL values. It treats early and late predictions symmetrically. Let $\widehat{RUL}_i$ be the predicted RUL for engine i, and RUL_i the true RUL. Then the RMSE is:

$$\text{RMSE} = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\widehat{RUL}_i - RUL_i\right)^2} \tag{5}$$

NASA Asymmetric Score (Saxena Score) [19] penalizes late predictions more heavily than early ones, due to the higher operational cost of delayed maintenance. It is defined as:

$$S = \sum_{i=1}^{N} \begin{cases} \exp\left(-\dfrac{\widehat{RUL}_i - RUL_i}{\alpha_1}\right) - 1, & \text{if } \widehat{RUL}_i < RUL_i \\ \exp\left(\dfrac{\widehat{RUL}_i - RUL_i}{\alpha_2}\right) - 1, & \text{if } \widehat{RUL}_i \geq RUL_i \end{cases} \tag{6}$$

where $\alpha_1 = 13$, $\alpha_2 = 10$.

Safety Classification Metric counts how many engines were predicted with an RUL that does not exceed the actual value. This represents the number of *safe* predictions, avoiding late (dangerous) failures:

$$\text{Safe} = \sum_{i=1}^{N} \mathbb{1}\left[\widehat{RUL}_i \leq RUL_i\right] \tag{7}$$

Equation 7 offers a binary, interpretable view of safety performance. It complements the Saxena score by explicitly quantifying how often the model avoids overestimating the remaining life.

All metrics above are computed after generating final RUL predictions from the models under evaluation and are used in Section 5 to compare the reproduced baseline, the proposed HAT model, and the statistical ensemble.

5 Results

Table 2 presents the results of our experiments. The top three rows list the SOTA results on the RUL prediction task. The last four rows show our experimental results. The first two are the results obtained with the Health-Aware Features, the last two are those obtained with HAT, with and without the attentional bias.

HAT (with bias) achieves the best performance (lowest RMSE of 10.95), outperforming the version without bias and all other models. Interestingly, in terms of Safety percentage, the Health-Aware Features outperform HAT and have the additional advantage of offering transparency.

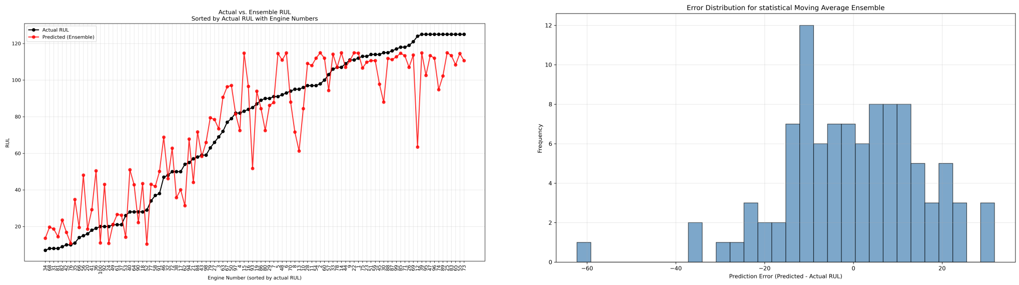
5.1 Details on Health-Aware Features Results

We examined the results obtained with both Health-Aware Features variants. Figure 6a shows the best performance with the moving-average variant. The graph displays RUL as a function of cycle, using the last 5-cycle moving average on the stabilized predictions and ensemble weights `[0.2, 0.8]` for Direct Mapping (DM) and Gradient Boosting (GB), respectively. The black curve shows the actual RUL (not strictly linear due to irregular sampling), while the red curve represents the predicted RUL.

Figure 6b presents the corresponding error distribution. Most errors lie between −20 and +20 cycles, with a central block near zero. The distribution is asymmetric, with several negative outliers below −60 cycles. These outliers explain the improved RMSE but lower Safety, with only 50 safe predictions.

Table 2: Overview of results on the RUL prediction task. Saxena scores and Safety percentages are not reported in prior works.

Model	Description	RMSE	Saxena Score	Safety %
GCU-Transformer [15]	Transformer + Gated Convolution	11.27	–	–
CTVAE [21]	Conv. Transformer + VAE	12.41	226.21	–
BiLSTM-Attention [7]	Bi-LSTM + Attention	13.21	320	–
Health-Aware Features (Moving Avg.)	Mahalanobis Distance (MD)	15.51	412	50
Health-Aware Features (Last Cycle)	Mahalanobis Distance (MD)	15.80	458	**53**
HAT without Attn. bias (Baseline, Ours)	GCU-Transformer	11.23	211.41	43
HAT (Ours)	GCU-Transformer + MD bias	**10.95**	**191.29**	38



(a) Actual and predicted RUL for Health-Aware Features with moving average (best RMSE).

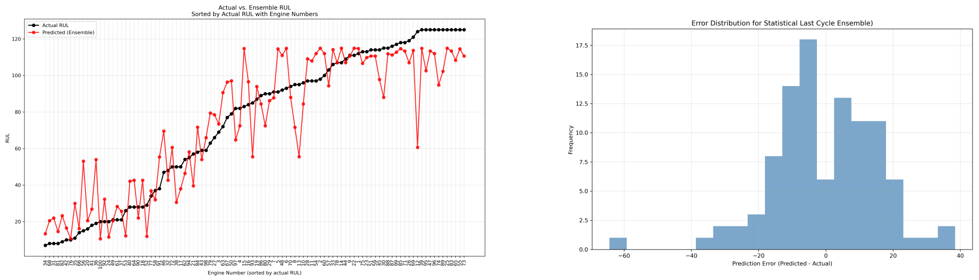
(b) Error distribution for Ensemble + Moving Average (DM = 0.2, GB = 0.8).

Fig. 6: Performance of Health-Aware Features with moving average. (a) RUL prediction trajectories. (b) Corresponding error distribution.

5.2 Base Strategy (Best Safety)

This strategy uses the final-cycle MD without smoothing, combined through ensemble weights `[0.1, 0.9]` for Direct Mapping (DM) and Gradient Boosting (GB) respectively.

Figure 7b shows the error distribution of this configuration. Errors are concentrated between −5 and +10 cycles, with fewer extreme outliers than in the Moving Average variant. This tighter distribution explains the higher Safety rate (53 safe predictions), despite a slightly higher RMSE.



(a) Actual and predicted RUL for Ensemble + Base strategy (best Safety).

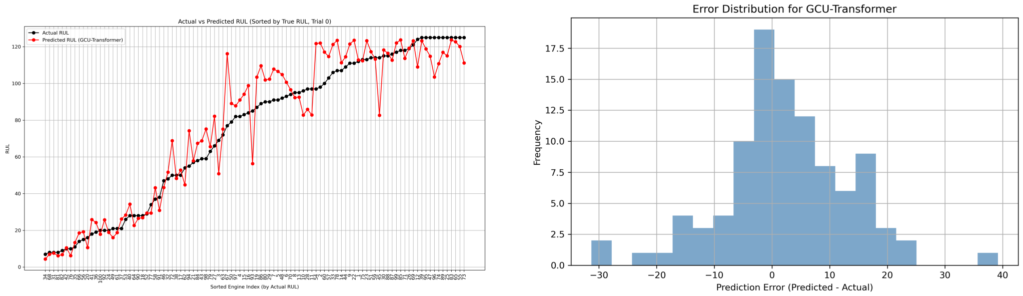
(b) Error distribution for Ensemble + Base strategy (DM = 0.1, GB = 0.9).

Fig. 7: Performance of Health-Aware Features with Base strategy. (a) RUL prediction trajectories. (b) Corresponding error distribution.

As highlighted in Figures 6b and 7b, the two ensemble strategies exhibit distinct error behaviors. The Moving Average ensemble reduces variance and improves RMSE but introduces more extreme negative outliers, lowering Safety. In contrast, the Base ensemble yields a narrower and more symmetric error distribution, achieving higher Safety at a minor increase in RMSE. This trade-off underscores the operational priority of safety, where avoiding late RUL predictions is critical to prevent unplanned failures.

5.3 GCU-Transformer Baseline

The reproduced baseline closely matches published results [15], achieving an RMSE of 11.23 and 43 safe predictions. Figure 8a shows the predicted and actual RUL, while Figure 8b presents the corresponding error distribution. Errors are more dispersed than in our methods, with outliers between −30 and +38 cycles, reflecting higher variance.



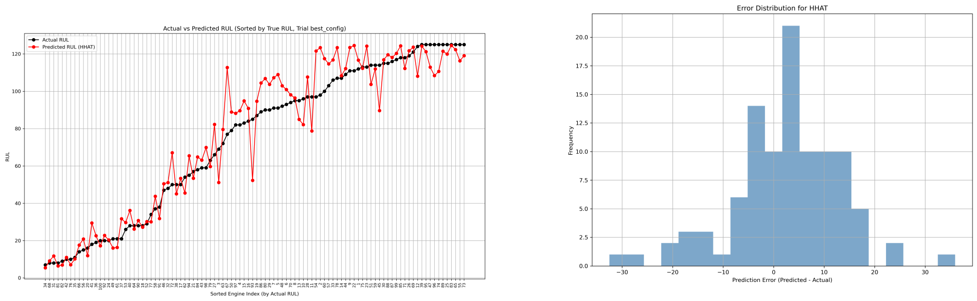
(a) Actual and predicted RUL.

(b) Error distribution.

Fig. 8: Performance of the reproduced GCU-Transformer baseline. (a) RUL prediction trajectories. (b) Corresponding error distribution.

5.4 Health-Aware Transformer (HAT)

HAT incorporates a health-aware bias into the attention mechanism. It achieves the lowest RMSE (10.95) and Saxena Score (191), but only 38 safe predictions. Figure 9a shows the predicted and actual RUL, while Figure 9b presents the error distribution. Compared to the baseline, the error spread is narrower and concentrated between −5 and +15 cycles, but this comes at the cost of fewer safe predictions.



(a) RUL prediction trajectories.

(b) Error distribution.

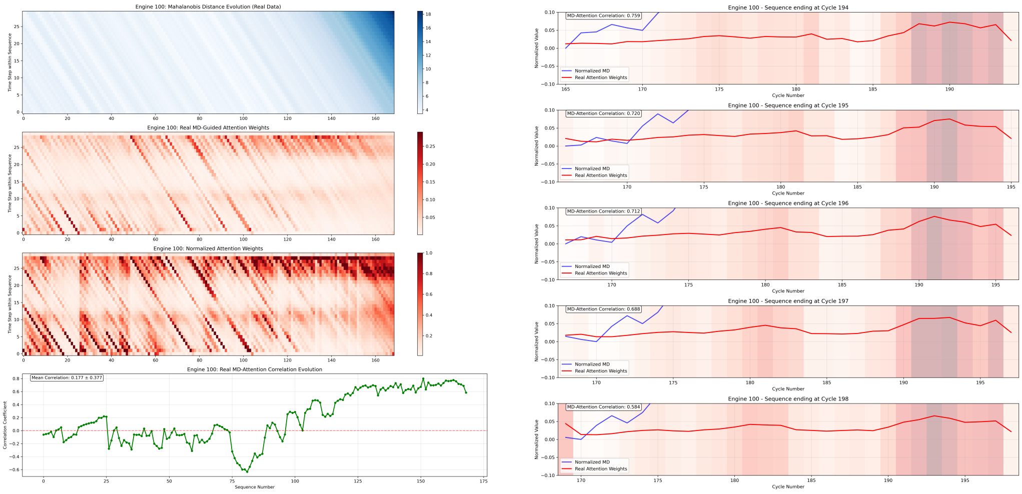
Fig. 9: Performance of the Health-Aware Transformer (HAT). (a) Actual vs. predicted RUL. (b) Corresponding error distribution.

To demonstrate the internal mechanism of HHAT, we extracted real attention weights from the trained model during inference. Figure 10(a) shows Engine 100 across multiple sequences, where normalized MD and learned attention weights exhibit moderate to strong positive correlations (r = 0.58-0.76). The attention mechanism demonstrates adaptive behavior, with correlations varying across engines and operational phases. Figure 10(b) provides a comprehensive view of Engine 100, showing how MD-attention correlations evolve throughout the engine's operational life. The correlation progression from near-zero to 0.7 indicates that the model progressively learns to associate attention patterns with degradation indicators, validating the effectiveness of our HAT's Health aware attention mechanism.

A high MD value means that the joint sensor vector shifts away from the healthy covariance structure. It reflects a collective change across correlated sensors, not a single anomaly. Thus, MD quantifies how normal inter-sensor relationships distort as degradation progresses, linking deviation strength to physical deterioration.

Encoder Depth Ablation To examine the influence of Transformer depth on Remaining Useful Life (RUL) prediction, HAT was trained with one to five encoder layers under identical settings. Table 3 summarizes the RMSE, Saxena Score, correlation, and safety percentage for each configuration. Performance degrades progressively with increased depth, showing that one encoder layer captures the relevant degradation dynamics most effectively once Mahalanobis Distance (MD) guidance is applied.

The results confirm that increasing the number of Transformer layers does not improve accuracy or stability. The one-layer configuration achieves the lowest RMSE and highest correlation while maintaining balanced safety performance. This indicates that the MD bias already embeds sufficient temporal degradation context, reducing the need for additional self-attention depth.



(a) MD-attention correlation across sequences.

(b) Correlation evolution over engine lifetime.

Fig. 10: Empirical validation of MD-guided attention. (a) Correlation strengthens as degradation progresses.(b) Real attention weights correlate with health indicators.

Table 3: Ablation on Transformer encoder depth for the FD001 dataset.

Layers	**RMSE**	**Saxena Score**	**Correlation**	**Safety (%)**
1	10.95	191.29	0.9655	38
2	11.91	216.63	0.9593	44
3	12.53	297.63	0.9562	38
4	13.09	314.41	0.9528	36
5	13.02	304.47	0.9519	44

5.5 Trade-offs and Discussion

Figures 8b and 9b show contrasting error patterns. The GCU-Transformer baseline has higher variance and wider error ranges, yet yields more safe predictions (43) than HAT (38). In contrast, the HAT model, which incorporates a health-aware attention bias, achieved a lower RMSE of 10.95 and a safety rate of 38%. Compared to the baseline, the error spread is narrower and concentrated between −5 and +15 cycles, but this comes at the cost of fewer safe predictions.

A significant architectural finding for HAT was that a single Transformer layer proved optimal; deeper stacks led to degraded performance, suggesting that the integrated health-aware bias already provides sufficient long-term degradation context. This implies that the MD bias effectively acts as a compact, external source of degradation memory, reducing the need for the Transformer's inherent sequential processing to learn long-term dependencies across multiple layers. This interaction highlights how combining statistical insights with deep learning can streamline model architecture and give it an interpretable characteristic.

These findings collectively demonstrate a clear trade-off: the GCU-Transformer baseline tends to produce safer outcomes by issuing more conservative predictions, while HAT reduces error but shifts predictions upward. This implies that the baseline model is preferable when operational safety is prioritized, whereas HAT is more suitable when maximizing predictive accuracy is the goal. The trade-off arises from model behavior. HAT, optimized for minimal error, tends to produce confident but sometimes late predictions near failure, whereas the statistical ensemble, using direct MD mappings, reacts earlier once deviations appear, leading to conservative estimates. The asymmetry in error distribution therefore reflects differing confidence and risk attitudes between the two approaches.

6 Conclusion

This study presents the Health-Aware Transformer (HAT), a novel approach that integrates statistical health indicators with deep learning architectures to advance both predictive accuracy and interpretability in Remaining Useful Life (RUL) prediction. Our primary contribution lies in developing and implementing health-aware features through Mahalanobis Distance (MD) and demonstrating how these can guide attention mechanisms in Transformer architectures.

The Health-Aware Transformer achieved state-of-the-art performance on NASA C-MAPSS FD001 with an RMSE of 10.95, surpassing reproduced GCU baseline best results (11.27 RMSE) while providing interpretable insights into model behavior. Through empirical analysis of real attention weights, we demonstrated that the health aware attention mechanism successfully learns to correlate with degradation patterns (r = 0.58-0.76), validating our approach's theoretical foundation.

A key strength of our methodology is its emphasis on explainability and interpretability. Unlike black-box deep learning approaches, HAT provides transparent health indicators through MD calculations and interpretable attention patterns that reveal how the model focuses on different degradation phases. The statistical ensemble approach further demonstrates that interpretable methods, while achieving higher error rates (RMSE 15.51-15.80), provide superior safety rates (50-53%) compared to deep learning models (38%), highlighting important trade-offs for safety-critical applications.

This work establishes the foundational framework for health-aware attention mechanisms. The current focus on FD001 (single operating condition, single fault mode) provides a controlled environment to validate the core concepts of the attention. The demonstrated success of integrating statistical health indicators with neural attention mechanisms opens clear pathways for extension to more complex scenarios.

Future work will naturally extend this framework to the remaining C-MAPSS datasets (FD002–FD004), which feature multiple operating conditions and concurrent fault modes. These datasets introduce varying operating regimes, requiring either separate healthy baselines per condition or a cluster-based adaptive covariance structure to maintain a meaningful Mahalanobis Distance interpretation. This progression will test the robustness and generalizability of health-aware attention mechanisms across diverse operational scenarios. Additional research directions include adapting the MD-based health indicators to handle increased sensor noise, missing data, and variable operating environments encountered in real-world industrial settings. Another promising direction is decomposing the Mahalanobis Distance into sensor-level contributions to identify which sensors or sensor groups drive deviations from the healthy baseline, enabling finer-grained interpretability.

A Appendix

The Appendix establishes and justifies the choice of the health states (baselines) for both training and test engines, ensuring that the subsequent health-aware features are grounded in a consistent definition of the healthy state.

A.1 Training-set Healthy Baseline Selection

Sensor values differ across engines due to conditions, manufacturing variation, and initial wear. To enable comparability, z-score normalization was applied engine-wise. Means and standard deviations were estimated from a healthy baseline segment. We compared baseline window lengths of 20%, 25%, and 30% by modeling each baseline as a multivariate Gaussian and analyzing the covariance structure. Table 4 summarizes the results across 100 engines.

Table 4: Training-set covariance statistics for different baseline windows (FD001, $n = 100$ engines).

Baseline	Trace mean	Trace std	Logdet mean	Condition mean
20%	41.73	5.84	-43.84	6.94×10^6
25%	42.30	5.54	-43.06	5.85×10^6
30%	42.66	4.85	-42.61	5.40×10^6

Interpretation of metrics. The covariance structure was assessed using three metrics: **Trace** (the sum of eigenvalues) measures the total variance across all sensors; **Log-determinant** (log of the product of eigenvalues) measures the generalized variance or volume of the covariance ellipsoid; **Condition number** (ratio of largest to smallest eigenvalue) indicates numerical stability, with large values reflecting ill-conditioned covariance matrices.

While each metric provides complementary information, we also examined the combined value of *trace mean + logdet mean* as a simple stability indicator. When this sum is negative, the covariance remains dominated by healthy-state variability. When it approaches zero or becomes positive, early degradation cycles begin to influence the baseline.

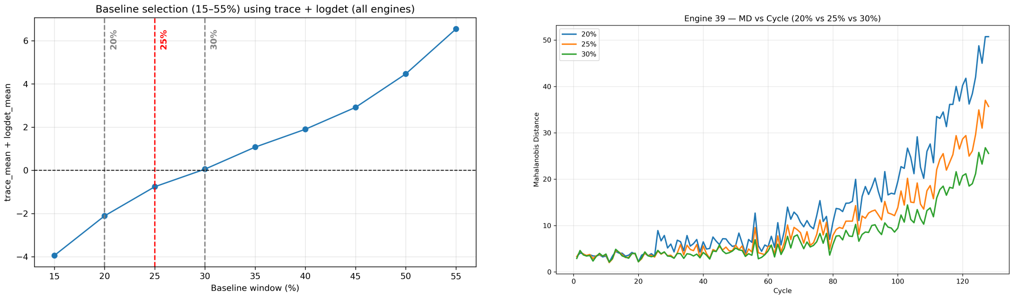
At 20%, the covariance matrix is less stable, as indicated by the more negative log-determinant (-43.84) and the higher condition number (6.94×10^6). At 30%, stability improves (-42.61, 5.40×10^6), but a critical observation arises: the combined measure of trace mean and log-determinant mean shifts sign between 25% and 30%. Up to 25%, the sum remains negative, reflecting a covariance structure dominated by healthy-state variability. At 30%, the sum becomes slightly positive, suggesting that early degradation cycles begin to influence the baseline statistics. To maintain precaution and avoid contamination by degradation, we therefore step back from 30% to 25%. This provides a balanced and conservative estimate of the healthy state, consistent with robust baseline selection practice.

Figure 11a illustrates this transition across all engines. The aggregated trend of trace and log-determinant shows that the 25% baseline lies just before the sign change at 30%, making the shift visually clear.

To further support this conclusion, Figure 11b shows Engine 39 as an example. Here the three baselines diverge: the 20% baseline inflates Mahalanobis distances, the 30% baseline delays the onset of degradation, and the 25% baseline provides a balanced trajectory between the two extremes.

A.2 Test-set Healthy Baseline Selection

For the test set, percentage-based baselines cannot be defined because the total lifetime is unknown. We therefore evaluated fixed baseline windows $W \in \{10, 20, 30, 40, 50, 60, 70, 80, 90, 100\}$ cycles. Table 5 reports the covariance statistics.



(a) Aggregated baseline statistics (trace + logdet, baselines 15–55%).

(b) Engine 39 MD trajectories under 20%, 25%, and 30% baselines.

Fig. 11: Baseline justification: (a) dataset-level stability, (b) engine-level example. The 25% baseline provides the most balanced definition of the healthy state.

Table 5: Test-set fixed-window covariance statistics (FD001, $n = 100$ engines). Engines with fewer than W cycles used all available cycles as baseline.

Cycles	Trace mean	Trace std	Logdet mean	Condition mean
10	39.65	9.74	-1.28×10^{2}	2.34×10^{13}
20	41.40	7.08	-4.82×10^{1}	1.89×10^{7}
30	42.14	5.97	-4.48×10^{1}	8.01×10^{6}
40	42.93	5.11	-4.36×10^{1}	6.49×10^{6}
50	43.29	4.79	-4.29×10^{1}	5.64×10^{6}
60	43.42	4.45	-4.25×10^{1}	5.23×10^{6}
70	43.81	4.07	-4.21×10^{1}	5.08×10^{6}
80	44.72	4.28	-4.18×10^{1}	4.95×10^{6}
90	45.41	4.58	-4.15×10^{1}	4.77×10^{6}
100	46.40	5.06	-4.13×10^{1}	4.76×10^{6}

Very short windows are unstable. At $W = 10$, the log-determinant is highly negative (-1.28×10^2) and the condition number reaches 2.34×10^{13}, indicating near-singular covariance. At $W = 20$, instability persists (-4.82×10^1, 1.89×10^7).

Stability emerges between 30 and 40 cycles. At $W = 30$, the log-determinant improves to -4.48×10^1 and the condition number drops to 8.01×10^6. At $W = 40$, the values further stabilize (-4.36×10^1, 6.49×10^6). Figure 12 shows this stabilization trend.

A clear knee appears at $W = 50$: the log-determinant aligns with the training baseline values (-4.29×10^1) and the condition number reaches 5.64×10^6. Beyond 50 cycles, further improvements are marginal, while the number of engines decreases (93 at 50 cycles versus 70 at 100).

Because $W = 50$ lies at this stability knee, is consistent with the training baseline length (≈80–90 cycles on average), and retains nearly all engines, we adopted 50 cycles as the fixed baseline for the test set. For engines with fewer than 50 cycles, all available cycles were used to define the baseline.

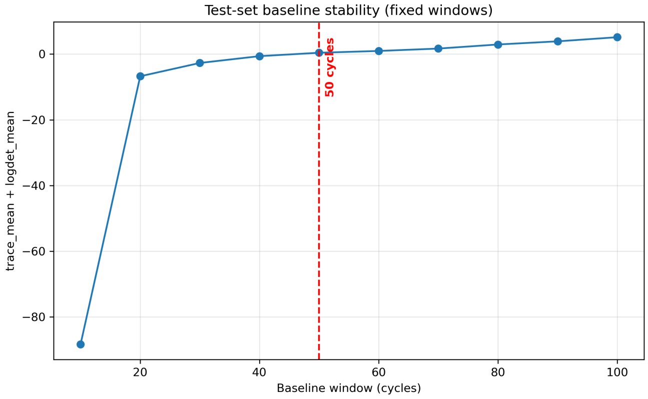


Fig. 12: Test-set aggregated baseline statistics across fixed windows ($W = 10$–100). A stability knee appears at $W = 50$.

Acknowledgments. This research was conducted at Jheronimus Academy of Data Science (JADS), Tilburg University, within the ILLUSTRE project.

Disclosure of Interests. The authors have no competing interests to declare that are relevant to the content of this article.

References

1. Chen, Z., Cao, S., Mao, Z.: Remaining useful life estimation of aircraft engines using a modified similarity and supporting vector machine (svm) approach. Energies **11**(1), 28 (2018). https://doi.org/10.3390/en11010028
2. Costa, R., Sánchez, L.: Variational autoencoder for interpretable remaining useful life estimation in industrial systems. In: Proceedings of the IEEE International Conference on Prognostics and Health Management (PHM). pp. 1–8. IEEE (2022)
3. Da Costa, P.R.D.O., Akcay, A., Zhang, Y., Kaymak, U.: Attention and long short-term memory network for remaining useful lifetime predictions of turbofan engine degradation. International journal of prognostics and health management **10**, 034 (2019)
4. Dashdondov, K., Kim, M.H.: Mahalanobis distance based multivariate outlier detection to improve performance of hypertension prediction. Neural Processing Letters **55**, 265–277 (2023). https://doi.org/10.1007/s11063-021-10663-y

5. De Maesschalck, R., Jouan-Rimbaud, D., Massart, D.: The mahalanobis distance. Chemometrics and Intelligent Laboratory Systems **50**(1), 1–18 (2000). https://doi.org/https://doi.org/10.1016/S0169-7439(99)00047-7, https://www.sciencedirect.com/science/article/pii/S0169743999000477
6. Fan, Y., Zhang, H., Li, W., Chen, X.: Bidirectional lstm autoencoder with transformer encoder for robust remaining useful life prediction. Reliability Engineering & System Safety **236**, 109315 (2023)
7. Fan, Z., Li, W., Chang, K.C.: A Bidirectional Long Short-Term Memory Autoencoder Transformer for Remaining Useful Life Estimation. Mathematics **11**(24), 4972 (Dec 2023). https://doi.org/10.3390/math11244972, https://www.mdpi.com/2227-7390/11/24/4972
8. Gopikrishna, V., Natarajan, R., Arvind, S.: Mahalanobis-taguchi system approach for remaining useful life prediction of equipment. In: 2005 IEEE Aerospace Conference. pp. 3655–3662. IEEE (2005). https://doi.org/10.1109/AERO.2005.1559697
9. Guo, L., Li, N., Jia, F., Lei, Y., Lin, J.: A recurrent neural network based health indicator for remaining useful life prediction of bearings. Neurocomputing **240**, 98–109 (2017). https://doi.org/https://doi.org/10.1016/j.neucom.2017.02.045, https://www.sciencedirect.com/science/article/pii/S0925231217303363
10. Kingma, D.P., Welling, M.: Auto-Encoding Variational Bayes (Dec 2022). https://doi.org/10.48550/arXiv.1312.6114, http://arxiv.org/abs/1312.6114, arXiv:1312.6114 [stat]
11. Lee, J., Kim, H., Kang, J., Lee, S.g.: Revisiting mahalanobis distance for transformer-based out-of-domain detection. arXiv preprint arXiv:2101.03778 (2021), https://arxiv.org/abs/2101.03778
12. Liu, K., Wang, J., Zhou, Y., Xu, Z.: Hybrid transformer-gru network for remaining useful life prediction of aero-engines. Mechanical Systems and Signal Processing **200**, 110565 (2023)
13. Liu, Z., Luo, M., Xie, Y., Chen, Z.: Remaining useful life prediction of the ball screw system based on weighted mahalanobis distance and an exponential model. Measurement Science and Technology **35**(1), 015108 (2023). https://doi.org/10.1088/1361-6501/adafc8
14. McLachlan, G.J.: Mahalanobis distance. Resonance **4**(6), 20–26 (1999)
15. Mo, Y., Wu, Q., Li, X., Huang, B.: Remaining useful life estimation via transformer encoder enhanced by a gated convolutional unit. Journal of Intelligent Manufacturing **32**(7), 1997–2006 (2021). https://doi.org/10.1007/s10845-021-01750-x, https://doi.org/10.1007/s10845-021-01750-x
16. Noot, J.P., Birmele, E., Rey, F.: LSTM and Transformers based methods for Remaining Useful Life Prediction considering Censored Data. PHM Society European Conference **8**(1), 10 (Jun 2024). https://doi.org/10.36001/phme.2024.v8i1.3974, https://papers.phmsociety.org/index.php/phme/article/view/3974
17. Santhosh, T., Gopika, V., Ghosh, A., Fernandes, B.: An approach for reliability prediction of instrumentation control cables by artificial neural networks and weibull theory for probabilistic safety assessment of npps. Reliability Engineering System Safety **170**, 31–44 (2018). https://doi.org/https://doi.org/10.1016/j.ress.2017.10.010, https://www.sciencedirect.com/science/article/pii/S0951832017312152
18. Sateesh Babu, G., Zhao, P., Li, X.L.: Deep convolutional neural network based regression approach for estimation of remaining useful life. In: Navathe, S.B., Wu, W., Shekhar, S., Du, X., Wang, X.S., Xiong, H. (eds.) Database Systems for Advanced Applications. pp. 214–228. Springer International Publishing, Cham (2016)
19. Saxena, A., Goebel, K., Simon, D., Eklund, N.: Damage propagation modeling for aircraft engine run-to-failure simulation. In: 2008 International Conference on Prognostics and Health Management. pp. 1–9 (2008). https://doi.org/10.1109/PHM.2008.4711414
20. Sekar, K., Shah, S.A., Antony Athithan, A., Mukil, A.: Role of machine learning approaches in remaining useful prediction: A review. In: Peng, S.L., Hsieh, S.Y., Gopalakrishnan, S., Duraisamy, B. (eds.) Intelligent Computing and Innovation on Data Science. pp. 361–370. Springer Nature Singapore, Singapore (2021)

21. Wang, J., Hu, X., Yang, Y.: Remaining useful life prediction method based on conv-transformer variational autoencoder. In: Proceedings of the 2024 5th International Conference on Artificial Intelligence and Electromechanical Automation (AIEA). pp. 900–908. IEEE, IEEE (2024). https://doi.org/10.1109/AIEA62095.2024.10692508
22. Wei, Y., Wu, D.: Conditional variational transformer for bearing remaining useful life prediction. Advanced Engineering Informatics **59**, 102247 (Jan 2024). https://doi.org/10.1016/j.aei.2023.102247, https://linkinghub.elsevier.com/retrieve/pii/S1474034623003750
23. Wu, H., Zhang, K., Li, T., Gao, F.: Deep learning for remaining useful life prediction: A survey of models, trends, and challenges. Journal of Manufacturing Systems **72**, 578–596 (2024)
24. Wu, H., Zhao, L., Zhang, N., Wu, G.: Ti-former: End-to-end useful life prediction model based on transformer-informer. In: 2024 IEEE 13th Data Driven Control and Learning Systems Conference (DDCLS). pp. 2170–2175. IEEE, Kaifeng, China (2024). https://doi.org/10.1109/DDCLS61622.2024.10606610
25. Wu, Y., Yuan, M., Dong, S., Lin, L., Liu, Y.: Remaining useful life estimation of engineered systems using vanilla lstm neural networks. Neurocomputing **275**, 167–179 (2018). https://doi.org/https://doi.org/10.1016/j.neucom.2017.05.063, https://www.sciencedirect.com/science/article/pii/S0925231217309505
26. Xiang, J., Chen, Y., Zhao, L.: Factorized fusion variational encoding for interpretable rul prediction. IEEE Transactions on Industrial Electronics **69**(11), 11545–11556 (2022)
27. Xiang, S., Nie, F., Zhang, C.: Learning a mahalanobis distance metric for data clustering and classification. Pattern Recognition **41**(12), 3600–3612 (2008). https://doi.org/https://doi.org/10.1016/j.patcog.2008.05.018, https://www.sciencedirect.com/science/article/pii/S0031320308002057
28. Yang, B., Liu, R., Zio, E.: Remaining useful life prediction based on a double-convolutional neural network architecture. IEEE Transactions on Industrial Electronics **66**(12), 9521–9530 (2019). https://doi.org/10.1109/TIE.2019.2924605
29. Yang, Z., Baraldi, P., Zio, E.: A comparison between extreme learning machine and artificial neural network for remaining useful life prediction. In: 2016 Prognostics and System Health Management Conference (PHM-Chengdu). pp. 1–7 (2016). https://doi.org/10.1109/PHM.2016.7819794
30. Zhang, J., Li, W., Guo, P.: Prediction reliability assessment based on mahalanobis distance and gru in the application of bearing rul analysis. Applied Sciences **15**(8), 4441 (2023). https://doi.org/10.3390/app15084441
31. Zhao, L., Zhu, Y., Zhao, T.: Deep learning-based remaining useful life prediction method with transformer module and random forest. Mathematics **10**(16), 2921 (2022). https://doi.org/10.3390/math10162921
32. Zhou, K., Liu, Y., Wang, J.: Remaining useful life prediction methodologies with health indicator dependence for rotating machinery: A comprehensive review. In: Intelligent Computing and Optimization, pp. 333–348. Springer (2025). https://doi.org/10.1007/978-3-031-26193-0_50