Scientific Journal for Publishing in Health Research and Technology

Volume 2, Issue 3, 2026, Pages: 145-161

Journal homepage: https://sjphrt.com.ly/index.php/sjphrt/en/index

Leakage-Resistant and Uncertainty-Aware Remaining Useful Life Prediction: An Engine-Disjoint Benchmark and Deployable Application on NASA C-MAPSS

Esam Miftah Abdulnabi Aboudoumat [1*], Nabeel Faraj Amhimmid Abdullah [2], Ashraf Faraj Saed Albarki [3], AbdelAziz Ibrahim Radwan Bader [4]

[1,2] Computer Department, College of Science and Technology Qumins, Qumins, Libya

[3] Department of Computer Science, Faculty of Arts and Sciences Qumins, University of Benghazi, Qumins, Libya

[4] Department of Computer Science, Higher Institute of Engineering Technologies, Benghazi, Libya

التنبؤ المقاوم لتسرّب البيانات والواعي بعدم اليقين بالعمر التشغيلي المتبقي: تقييم مفصول حسب المحرك وتطبيق قابل للنشر باستخدام بيانات *NASA C-MAPSS*

عصام مفتاح عبد النبي بودومات[1*]، نبيل فرج امحمد عبدالله [2]، أشرف فرج سعد البركي[3]، عبد العزيز إبراهيم رضوان بدر[4]

[1،2] قسم الحاسوب، كلية العلوم والتقنية قمينس، قمينس، ليبيا

[3] قسم علوم الحاسوب، كلية الآداب والعلوم قمينس، جامعة بنغازي، قمينس، ليبيا

[4] قسم علوم الحاسوب، المعهد العالي للتقنيات الهندسية، بنغازي، ليبيا

[*]Corresponding author: esam.mouftah@gmail.com

Received: May 20, 2026	Accepted: July 25, 2026	Published: August 15, 2026

Abstract:

Remaining useful life (RUL) models are frequently ranked by point error while two deployment-critical questions remain unresolved: whether evaluation isolates complete assets and whether the reported prediction is accompanied by calibrated uncertainty. We present an engine-disjoint and leakage-resistant benchmark on all four NASA C-MAPSS subsets, together with a bilingual research application. Causal features use only the current and preceding cycles. Complete engine identities are separated during grouped validation and conformal calibration. Ridge regression, histogram gradient boosting (HGB), random forests, and extremely randomized trees are evaluated at the official test endpoint. The best RMSE values are 19.112, 27.895, 17.989, and 28.298 cycles on FD001–FD004, respectively. No single feature/model combination dominates all subsets. At nominal 90% coverage, split-conformal intervals attain 0.990, 0.892, 1.000, and 0.948 coverage, whereas conformalized quantile regression attains 0.960, 0.741, 1.000, and 0.915. Conditional analysis exposes severe CQR undercoverage for FD002 engines whose true RUL exceeds 90 cycles. Sensor corruption further reveals that a fixed calibration layer is not shift-proof: 5% standardized noise raises RMSE from 27.80 to 39.33 on FD002 and from 29.16 to 45.61 on FD004, while split-conformal coverage falls to 0.757 and 0.819. We therefore recommend reporting engine-level

separation, marginal and conditional coverage, and corruption stress tests as a minimum reliability protocol. The supplied application returns a point estimate, two 90% intervals, and an explicit research-use warning. All data, code, trained artifacts, figures, and numerical tables are provided for reproduction.

Keywords: remaining useful life; prognostics and health management; C-MAPSS; conformal prediction; uncertainty quantification; grouped validation; data leakage; robustness.

الملخص

يُرتَّب كثير من نماذج العمر التشغيلي المتبقي وفق خطأ التنبؤ النقطي فقط، مع إهمال فصل الأصول كاملةً ومعايرة عدم اليقين. نقدّم تقييماً مقاوماً لتسرّب البيانات ومفصولاً حسب المحرك على مجموعات NASA C-MAPSS الأربع، مع تطبيق بحثي ثنائي اللغة. تعتمد الخصائص على الدورة الحالية وما سبقها فقط، ولا ينتقل أي محرك بين التدريب والمعايرة والتحقق. بلغت أفضل قيم RMSE مقدار 19.112 و27.895 و17.989 و28.298 دورة في FD001–FD004. وعند تغطية اسمية 90%، حققت المطابقة المقسّمة تغطية 0.990 و0.892 و1.000 و0.948، بينما حقق الانحدار الكمي المطابق 0.960 و0.741 و1.000 و0.915. أظهر تحليل الشرائح نقصاً واضحاً في تغطية FD002 عند RUL أكبر من 90 دورة. كما رفعت ضوضاء معيارية بنسبة 5% خطأ FD002 من 27.80 إلى 39.33 وFD004 من 29.16 إلى 45.61. تؤكد النتائج ضرورة الجمع بين الفصل على مستوى المحرك، والتغطية الهامشية والشرطية، واختبارات تغيّر الحساسات قبل التفكير في الاستخدام التشغيلي.

الكلمات المفتاحية: العمر المتبقي المفيد؛ التشخيص والتنبؤ بالحالة الصحية؛ C-MAPSS؛ التنبؤ المطابق؛ تحديد كمية عدم اليقين؛ التحقق المجمع؛ تسرب البيانات؛ المتانة.

1. Introduction

Remaining useful life is the estimated number of operating cycles before an asset reaches a defined end-of-life condition. Reliable RUL estimation supports inspection planning, spare-parts allocation, and maintenance prioritization. The task is difficult because degradation is only partially observed, operating regimes vary, and failure labels are scarce. Reviews consistently distinguish data-driven, physics-based, and hybrid prognostics, and they stress that performance must be interpreted in relation to the available sensing and failure mechanism [9]–[11].

C-MAPSS remains a central benchmark because it contains run-to-failure trajectories with controlled changes in operating conditions and fault modes [1]–[3]. Recurrent networks, convolutional networks, and semi-supervised architectures have substantially advanced point prediction on these data [4]–[8]. However, a low RMSE is not, by itself, an operational guarantee. Random row splitting can place cycles from the same engine in both training and validation; future-looking feature construction can expose unavailable information; and an interval may have acceptable average coverage while failing in a critical RUL region. These problems concern scientific validity rather than model capacity.

Uncertainty-aware prognosis has therefore become a distinct research direction [12], [18]–[21]. Conformal prediction is attractive because it can wrap a fitted regressor and provide finite-sample marginal coverage under exchangeability [13]–[16]. The same assumption is also a boundary: distribution shift can invalidate the guarantee [17]. A useful benchmark must expose that boundary instead of presenting an interval as universally calibrated.

1.1 Research Questions

RQ1: How do transparent linear and tree-ensemble regressors compare at the official C-MAPSS test endpoint when complete engine identities are kept disjoint?

RQ2: Do split-conformal and conformalized-quantile intervals achieve their nominal 90% coverage, both marginally and across true-RUL bands?

RQ3: How do point accuracy and interval validity change under controlled sensor noise and missingness?

RQ4: Can the trained pipeline be packaged as a reproducible research application without overstating deployment readiness?

1.2 Contributions

We make four concrete contributions.

A leakage-resistant protocol that uses causal features, engine-grouped validation, engine-level conformal calibration, and the untouched official test engines.

A complete four-subset comparison of snapshot and temporal representations across five regressors plus a fleet-life baseline, with engine bootstrap intervals and paired nonparametric tests.

A reliability audit that reports marginal coverage, conditional coverage, interval width, Winkler score, and corruption-induced degradation.

A bilingual local web/command-line application and a reproducibility package containing the official dataset archive, checksum, source code, trained models, tables, and figures.

2. Related Work

2.1 Point RUL Prediction on C-MAPSS

The original simulator generated multivariate trajectories under specified operational and degradation scenarios [2]. Subsequent benchmarking clarified that FD001 and FD003 have one operating condition, whereas FD002 and FD004 include six; FD003 and FD004 also contain two fault modes [3]. This heterogeneity is important: an algorithm that works well on a single-condition subset may not transfer to multi-condition data.

Sequence models learn degradation patterns directly from ordered measurements. Heimes used recurrent neural networks [4]; Babu et al. proposed a deep convolutional regression model [5]; Zheng et al. applied long short-term memory networks [6]; and Li et al. developed a deep convolutional approach for prognostics [7]. Semi-supervised architectures have also exploited unlabeled structure [8]. These methods demonstrate the value of temporal representation, but the literature also contains wide variation in target capping, preprocessing, sample construction, and validation. Ramasso and Saxena consequently argued for explicit protocol reporting when comparing C-MAPSS results [3].

We deliberately use compact tabular ensembles rather than a larger neural architecture. Gradient boosting, random forests, and extremely randomized trees are established nonlinear learners [22]–[24], while scikit-learn provides reproducible implementations [25]. This choice keeps the practical model artifacts below 1.1 MB and shifts attention from architecture novelty to evaluation reliability.

2.2 Uncertainty Quantification

Predictive uncertainty combines irreducible outcome variability and uncertainty induced by incomplete knowledge [18]. Deep Gaussian processes and probabilistic deep-learning methods have been proposed for RUL applications [19], [20], and recent benchmarking confirms that uncertainty quality must be evaluated separately from point accuracy [21]. An interval is informative only when its coverage and sharpness are jointly reported.

Split conformal prediction calibrates the residuals of an arbitrary point model on held-out exchangeable examples [13], [15], [16]. Conformalized quantile regression (CQR) instead begins with conditional quantile estimates and conformalizes their nonconformity scores [14]. CQR can adapt interval width to the input, but finite-sample marginal validity still depends on

the relationship between calibration and future cases. Conformal methods have previously been studied for RUL estimation [12]. Our comparison focuses on empirical behavior at the official engine endpoint and under sensor shift.

2.3 Leakage, Grouped Validation, and Shift

Each C-MAPSS engine contributes many correlated cycles. A row-level split therefore violates the independence implicit in ordinary cross-validation because measurements from the same physical unit can cross folds. Group-aware evaluation is the natural correction for hierarchical data [26]. A second leakage route arises when a rolling statistic is centered or otherwise uses future cycles. We prevent both routes: complete engines define folds, and every temporal statistic is strictly causal.

Conformal validity does not remove the shift problem. Barber et al. show why predictive inference beyond exchangeability requires additional treatment [17]. We therefore treat corruption tests as stress tests, not as new calibrated test distributions. A fall in coverage under noise is evidence that static calibration is insufficient for that shift.

3. Materials and Methods

3.1 Dataset and Outcome

We downloaded the official NASA C-MAPSS archive and verified its SHA-256 digest as c9c5dec12a945a82e8bb4446589d7fb3cc057b5e5d81fa1a12e25ee9912ad3b2. Each row contains an engine identifier, cycle index, three operating settings, and 21 sensor measurements [1]. Training trajectories reach simulated failure. Test trajectories stop earlier, and NASA provides the true RUL after the final observed cycle. The practical endpoint evaluation consequently produces one prediction per test engine.

Table 1. Verified characteristics of the four-official C-MAPSS subsets.

Subset	Train rows	Train engines	Test rows	Test engines	Conditions	Fault modes
FD001	20,631	100	13,096	100	1	1
FD002	53,759	260	33,991	259	6	1
FD003	24,720	100	16,596	100	1	2
FD004	61,249	249	41,214	248	6	2

Table 2. Dynamic channels, engineered features, and ranges in operating cycles.

Subset	Dynamic channels	Temporal features	Train life range	Test RUL range
FD001	17	69	128–362	7–145
FD002	24	97	128–378	6–194
FD003	18	73	145–525	6–145
FD004	24	97	128–543	6–195

The extracted FD004 files contain 249 training and 248 test engines. We report the counts observed in the downloaded archive rather than copying older descriptive text. This distinction is recorded in the manifest and can be checked directly from the included raw files.

3.2 Target Definition

For training engine i at cycle t, raw RUL is the difference between its terminal cycle T_i and the current cycle. We cap the training target at 125 cycles, a common stabilization that treats early life as a plateau. Evaluation always uses the uncapped NASA RUL.

$y_i,raw(t) = T_i - t, \qquad y_i(t) = \min(y_i,raw(t), 125). \qquad (1)$

Predictions are clipped at zero but not capped above 125 during evaluation. The distinction avoids negative lifetimes while retaining the full error against high test RUL values.

3.3 Causal Feature Construction

Channels with training standard deviation no greater than 10^{-8} are removed separately for each subset. The snapshot representation contains cycle plus the remaining operating settings and sensors. The temporal representation adds trailing means over 5 and 20 cycles and a five-cycle slope for each retained channel. For channel x, the features at cycle t are:

$\mu_i,w(t) = (1 / |W_i,w(t)|) \Sigma_k \in W_i,w(t)\ x_i(k), \quad W_i,w(t) = \{\max(1,t-w+1),\ldots,t\}, \qquad (2)$

$\Delta_i,5(t) = [x_i(t) - x_i(\max(1,t-5))] / \max(1,\min(5,t-1)). \qquad (3)$

Equations (2) and (3) exclude future cycles by construction. Median imputation is fitted within each model's training data. Ridge additionally standardizes its inputs. Depending on the subset, 17–24 dynamic raw channels survive and the temporal matrix contains 69–97 features.

3.4 Models

The fleet-life baseline predicts the mean terminal life of the training engines minus the observed test cycles. The learned comparators are Ridge regression with $\alpha = 10$; histogram gradient boosting (HGB); random forest (RF); and extremely randomized trees (ExtraTrees). HGB uses learning rate 0.06, 260 iterations, 31 leaf nodes, minimum leaf size 24, and L2 regularization 1.0. RF uses 160 trees, maximum depth 22, minimum leaf size 3, and 0.65 feature sampling. ExtraTrees uses 180 trees, minimum leaf size 2, and 0.80 feature sampling. Parameters were fixed before inspecting the official test labels; we did not optimize on the test set.

Table 3. Experimental configuration.

Component	Decision
Random seed	42 for splitting, fitting, corruption, and bootstrap streams
Training target	Piecewise-linear RUL capped at 125 cycles
Official test unit	One final observed row per engine; NASA-supplied true RUL
Validation	Five-fold GroupKFold by complete engine identity
Conformal split	75% fitting engines; 25% calibration engines
Calibration endpoint	One deterministic cutoff per calibration engine, sampled from 40%–95% of life
Nominal interval coverage	0.90 ($\alpha = 0.10$)
Bootstrap	2,000 engine-level resamples; percentile 95% CI
Software	Python 3.12.13; NumPy 2.3.5; pandas 2.2.3; SciPy 1.17.0; scikit-learn 1.8.0

3.5 Engine-Disjoint Evaluation

Five-fold Group KFold uses engine identity as the group. For each held-out engine, one deterministic pseudo-endpoint is selected between 40% and 95% of its trajectory. The model trains on all cycles from the other engines and is scored once per held-out engine against raw RUL. This design approximates the official endpoint task while avoiding repeated correlated

scores from the same held-out asset. We then fit each point model on all official training trajectories and evaluate the final available row of each official test engine.

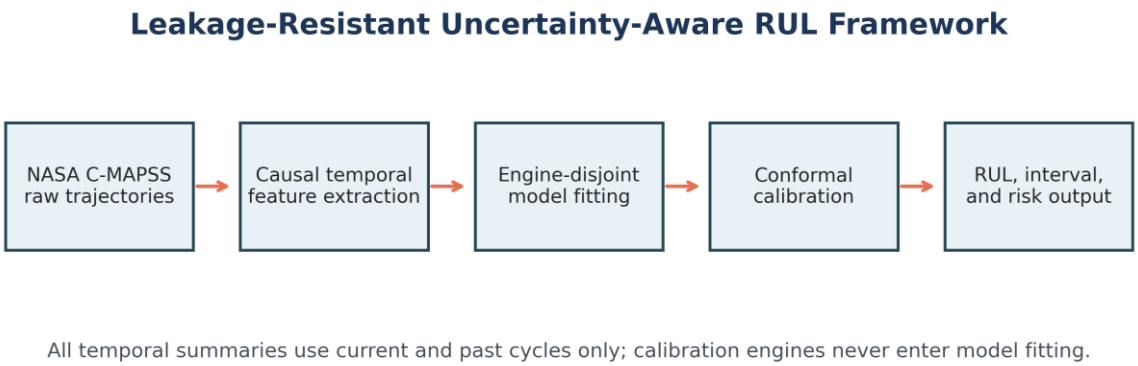

Figure 1. Leakage-resistant workflow from official trajectories to engine-disjoint testing and uncertainty audit.

3.6 Split-Conformal and CQR Intervals

For interval construction, complete training engines are randomly divided into 75% fitting and 25% calibration groups. Each calibration engine contributes one deterministic pseudo-endpoint, so the engine—not an individual cycle—is the exchangeability unit. For split conformal, the nonconformity score is $s_j = |y_j - \hat{y}_j|$. With n calibration engines and $\alpha = 0.10$, we use the finite-sample order statistic:

$$\hat{q} = s(\lceil (n+1)(1-\alpha) \rceil), \qquad C(x) = [\max(0, \hat{y}(x)-\hat{q}), \hat{y}(x)+\hat{q}]. \qquad (4)$$

For CQR, two HGB models estimate the 0.05 and 0.95 conditional quantiles. The calibration score is $\max\{\hat{q}_{0.05}(x_j)-y_j, y_j-\hat{q}_{0.95}(x_j)\}$; its conformal quantile expands both sides of the base interval [14]. The two methods share the same fit/calibration engines.

3.7 Metrics and Statistical Analysis

Point performance is measured with RMSE, MAE, R^2, mean signed bias, and the asymmetric NASA score [28]. Let $e_i = \hat{y}_i - y_i$. The score applies $\exp(-e_i/13)-1$ for $e_i < 0$ and $\exp(e_i/10)-1$ otherwise; overestimation is penalized more strongly. Because the score is a sum, it should be compared within the same subset, not across subsets with different engine counts.

$$\mathrm{RMSE} = \sqrt{[(1/n)\,\Sigma_i(\hat{y}_i-y_i)^2]}, \qquad \mathrm{MAE} = (1/n)\,\Sigma_i|\hat{y}_i-y_i|. \qquad (5)$$

Interval quality is reported as empirical coverage, mean prediction-interval width (MPIW), median width, and the Winkler interval score [27]. Lower width is desirable only when coverage remains adequate; the Winkler score penalizes both width and misses. We calculate exact two-sided Wilcoxon signed-rank tests on paired per-engine absolute errors [29]. Engine-level bootstrap confidence intervals use 2,000 resamples and the percentile method. No multiplicity-adjusted confirmatory claim is made.

3.8 Robustness Protocol

We evaluate the conformal HGB pipeline on four corrupted versions of each official test sequence: Gaussian sensor noise with standard deviation equal to 5% of each sensor's training standard deviation; independent 10% cell dropout; independent 20% dropout; and combined 5% noise plus 10% dropout. Operating settings remain unchanged. Missing values are handled

by the fitted median imputer. Crucially, intervals are not recalibrated after corruption. The experiment therefore measures the failure of a static deployment artifact under covariate shift.

3.9 Research Application

The supplied prototype accepts a raw 26-column C-MAPSS sequence for one engine, validates that at least five cycles are present, reconstructs the same causal features, loads the matching trained artifact, and returns the point prediction, split-conformal interval, CQR interval, and a non-authoritative risk tier. A command-line interface supports batch use; a local web interface provides Arabic and English labels. Files are processed in memory and are not uploaded. The interface explicitly states that the output is for research and education, not certified aviation maintenance.

4. Results

4.1 Point Prediction

Table 4. Complete endpoint point-prediction results on the official test engines; lower is better except R^2.

Subset	Model	RMSE	MAE	R^2	NASA score
FD001	Fleet-life baseline	37.778	28.320	0.174	25,970.4
FD001	Ridge-Snapshot	22.553	17.786	0.705	1,301.9
FD001	HGB-Snapshot	19.190	14.734	0.787	656.1
FD001	RF-Temporal	19.112	13.919	0.788	661.8
FD001	ExtraTrees-Temporal	19.379	13.981	0.783	683.4
FD001	HGB-Temporal	19.821	14.138	0.772	763.1
FD002	Fleet-life baseline	41.037	33.042	0.418	114,221
FD002	Ridge-Snapshot	31.905	24.050	0.648	15,719.3
FD002	HGB-Snapshot	27.895	19.955	0.731	9,459.5
FD002	RF-Temporal	28.043	20.032	0.728	9,852.4
FD002	ExtraTrees-Temporal	28.843	21.472	0.712	9,034.2
FD002	HGB-Temporal	27.929	19.863	0.730	9,846.7
FD003	Fleet-life baseline	45.544	34.475	-0.210	102,641
FD003	Ridge-Snapshot	22.563	18.179	0.703	1,719.3
FD003	HGB-Snapshot	18.693	14.156	0.796	849.9
FD003	RF-Temporal	18.968	14.226	0.790	823.8
FD003	ExtraTrees-Temporal	19.043	14.189	0.788	914.1
FD003	HGB-Temporal	17.989	13.449	0.811	606.8
FD004	Fleet-life baseline	58.895	47.319	-0.167	2,947,505
FD004	Ridge-Snapshot	36.138	29.226	0.561	16,526.9
FD004	HGB-Snapshot	28.298	21.176	0.731	5,291.1
FD004	RF-Temporal	29.273	21.865	0.712	6,731.0
FD004	ExtraTrees-Temporal	31.553	24.653	0.665	9,431.6
FD004	HGB-Temporal	28.603	21.491	0.725	5,899.3

The fleet-life baseline is consistently weak, confirming that observed cycles alone do not explain degradation state. The lowest test RMSE is obtained by RF-Temporal on FD001 (19.112), HGB-Snapshot on FD002 (27.895), HGB-Temporal on FD003 (17.989), and HGB-Snapshot on FD004 (28.298). The ranking changes across subsets. On FD002, ExtraTrees has

a lower NASA score than the HGB variants despite a higher RMSE, illustrating that point metrics encode different operational preferences.

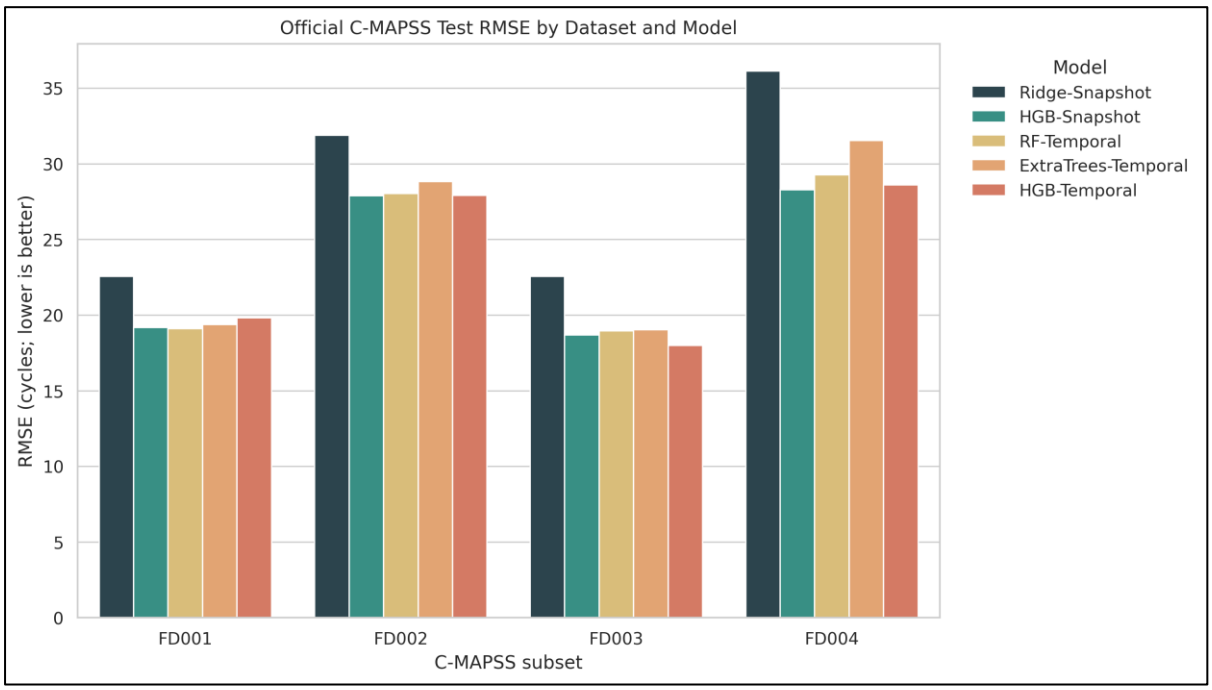

Figure 2. Official test RMSE by subset and learned model; the fleet-life baseline is omitted for scale.

The observed-versus-predicted plots show that errors are not uniform across the RUL range. High-RUL engines are often pulled toward the 125-cycle training plateau, while some low-RUL cases remain difficult in the multi-condition sets. This is a practical reason to complement global RMSE with conditional diagnostics.

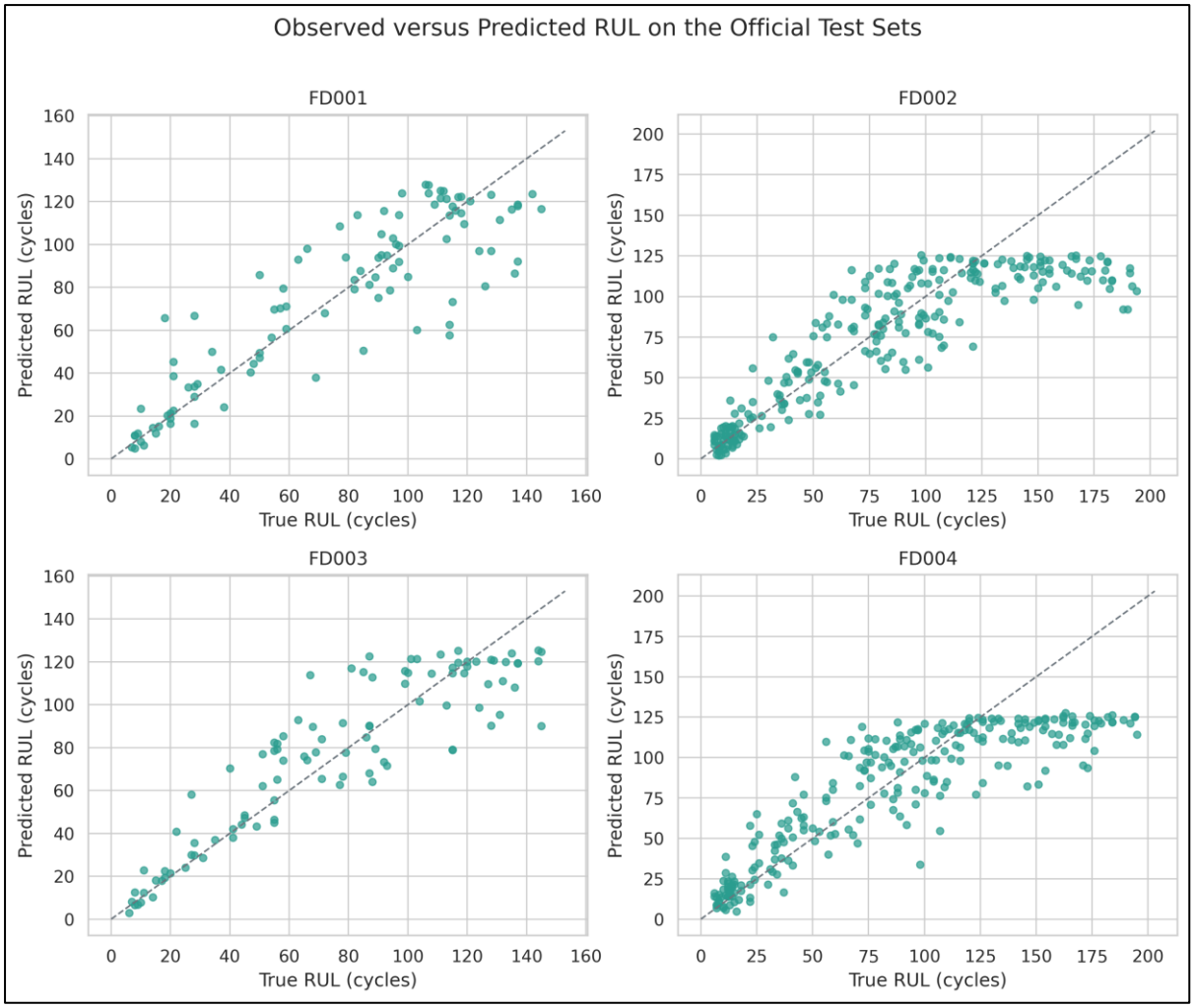

Figure 3. HGB-Temporal endpoint predictions against official true RUL; the dashed line is perfect agreement.

4.2 Grouped Validation and Representation Effects

Table 5. Mean ± standard deviation for temporal HGB under five-fold engine-grouped validation.

Subset	RMSE	MAE	R^2
FD001	21.75 ± 4.40	16.13 ± 3.08	0.609 ± 0.164
FD002	25.01 ± 2.11	18.80 ± 1.66	0.554 ± 0.125
FD003	36.98 ± 5.04	23.66 ± 4.17	0.491 ± 0.088
FD004	31.31 ± 4.35	21.34 ± 1.51	0.523 ± 0.154

Grouped validation is more variable than a single leaderboard number. Mean temporal-HGB RMSE ranges from 21.75 on FD001 to 36.98 on FD003, with fold standard deviations of 2.11–5.04 cycles. A separate, fold-matched representation check finds that temporal features improve mean RMSE only on FD001 (22.85 to 21.75). Snapshot HGB is slightly better in grouped validation on FD002 (24.70 versus 25.01), FD003 (35.22 versus 36.98), and FD004 (30.91 versus 31.31). The official FD003 test set nevertheless favors temporal HGB. Thus, causal temporal summaries are useful but not uniformly superior.

Table 9. Two-sided Wilcoxon tests comparing snapshot and temporal HGB per-engine absolute errors.

Subset	p-value	Median difference	Significant at 0.05
FD001	0.5822	-0.243	No
FD002	0.7315	0.084	No
FD003	0.5405	0.196	No
FD004	0.1563	-0.104	No

None of the snapshot-versus-temporal HGB comparisons is significant at 0.05. ExtraTrees has significantly larger paired absolute errors than temporal HGB on FD002 ($p = 0.0016$) and FD004 ($p < 0.0001$); the median differences are 0.85 and 2.05 cycles. We treat these tests as descriptive because several models and subsets are inspected.

4.3 Prediction Intervals

Table 6. Prediction-interval quality at nominal 0.90 coverage.

Subset	Method	Cal. engines	Coverage	MPIW	Winkler
FD001	Split conformal	25	0.990	87.85	90.19
FD001	Conformalized quantile regression	25	0.960	76.22	82.36
FD002	Split conformal	65	0.892	78.77	123.37
FD002	Conformalized quantile regression	65	0.741	82.88	218.67
FD003	Split conformal	25	1.000	108.10	108.10
FD003	Conformalized quantile regression	25	1.000	155.59	155.59
FD004	Split conformal	63	0.948	117.83	124.25
FD004	Conformalized quantile regression	63	0.915	144.49	167.20

Split conformal is the more reliable default in this experiment. It reaches or nearly reaches 0.90 coverage on every subset: 0.990, 0.892, 1.000, and 0.948. Its intervals are conservative on FD001, FD003, and FD004, partly because only 25 or 63–65 calibration engines define a high finite-sample quantile. CQR is narrower on FD001 but much wider on FD003 and FD004. Most importantly, its FD002 coverage is only 0.741 and its Winkler score is 218.67, revealing many costly misses.

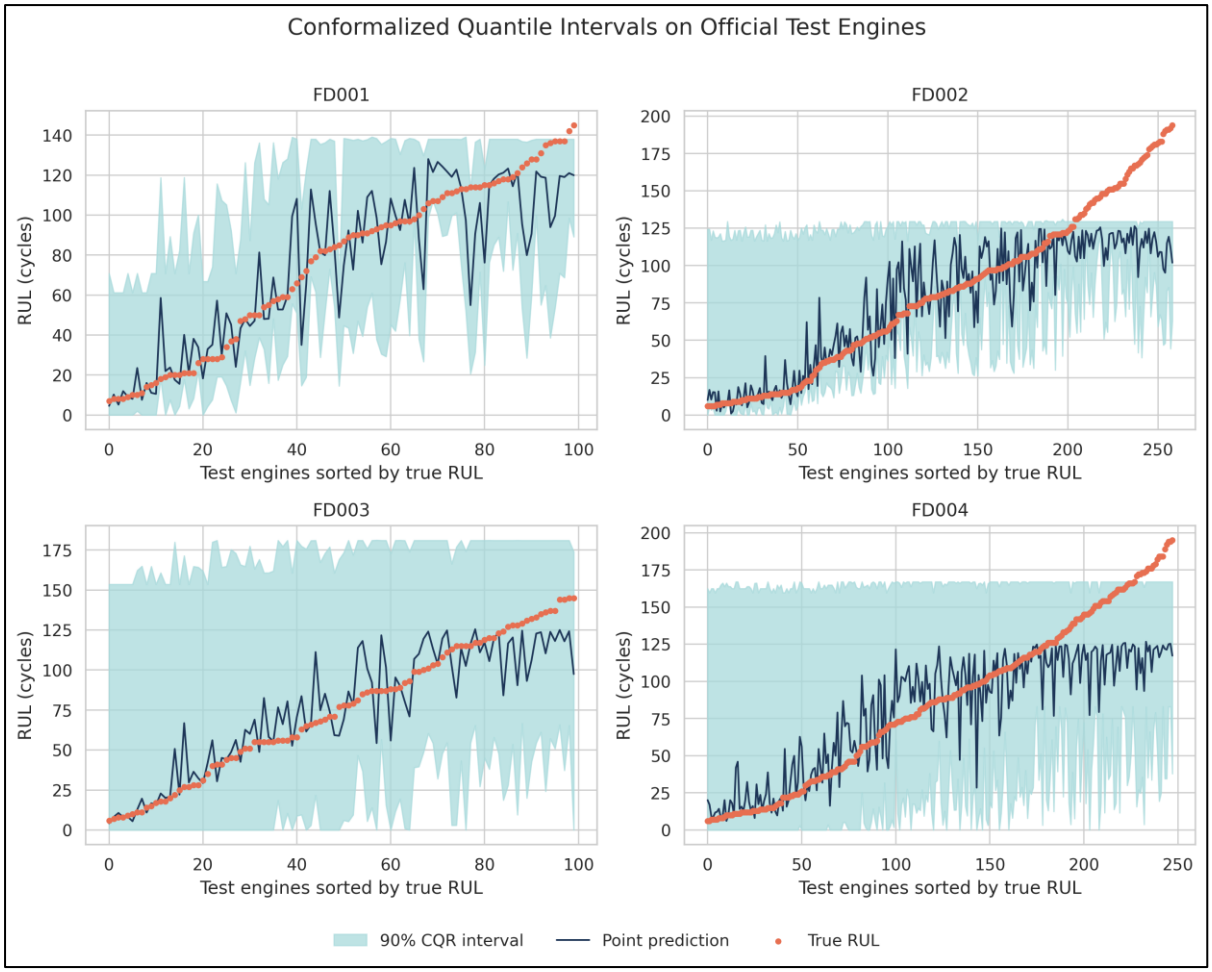

Figure 4. CQR intervals on official test engines sorted by true RUL; visual misses are concentrated in difficult regions.

Marginal coverage can conceal conditional failure. When engines are stratified by true RUL, CQR coverage on FD002 is 0.951 for 0–30 cycles, 0.976 for 31–60, 0.915 for 61–90, and only 0.464 above 90 cycles. FD004 also drops to 0.821 above 90. The result does not contradict marginal conformal theory because the official test truncation distribution need not be exchangeable with our pseudo-endpoint calibration sample, and marginal validity does not imply equal conditional coverage.

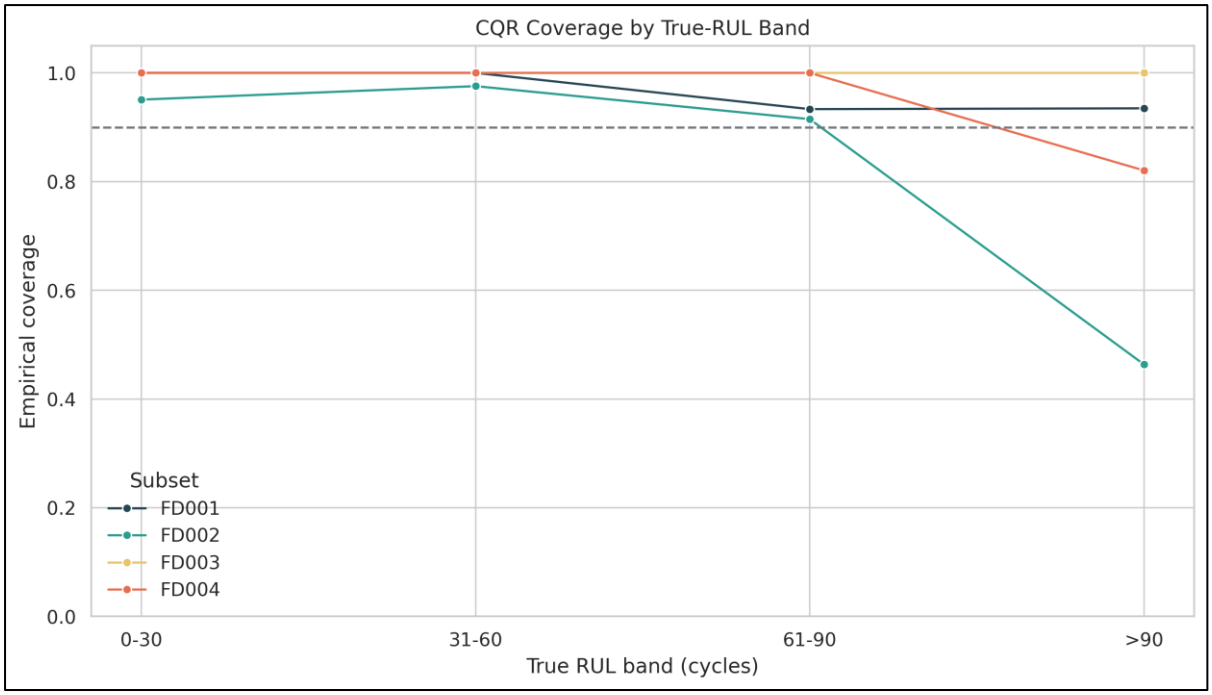

Figure 5. CQR coverage by true-RUL band; the dashed line marks nominal 0.90 coverage.

4.4 Uncertainty of the Error Estimates

Table 8. Engine-level bootstrap confidence intervals for temporal-HGB RMSE.

Subset	RMSE	95% lower	95% upper	Replicates
FD001	19.82	16.50	23.00	2000
FD002	27.93	24.73	31.08	2000
FD003	17.99	15.33	20.59	2000
FD004	28.60	25.86	31.46	2000

The 95% engine-bootstrap intervals show the sampling uncertainty behind each test RMSE. The interval is [16.50, 23.00] on FD001, [24.73, 31.08] on FD002, [15.33, 20.59] on FD003, and [25.86, 31.46] on FD004. Overlap should not be interpreted as a formal cross-subset comparison because the subsets represent different operating and fault configurations.

4.5 Sensor-Corruption Stress Test

Table 7. Robustness audit; noise equals 5% of training sensor standard deviation and dropout is cellwise.

Subset	Scenario	RMSE	MAE	Split coverage	Winkler
FD001	clean	18.79	13.56	0.990	90.2
FD001	noise_05	19.31	13.86	0.980	90.6
FD001	dropout_20	18.97	13.33	0.990	91.4
FD001	combined	20.12	14.19	0.970	92.1
FD002	clean	27.80	19.88	0.892	123.4
FD002	noise_05	39.33	29.40	0.757	198.9

FD002	dropout_20	31.80	23.50	0.849	144.6
FD002	combined	40.19	30.26	0.749	204.4
FD003	clean	17.13	12.56	1.000	108.1
FD003	noise_05	17.46	13.09	1.000	107.8
FD003	dropout_20	18.55	13.98	1.000	108.4
FD003	combined	17.51	12.85	1.000	108.1
FD004	clean	29.16	21.93	0.948	124.3
FD004	noise_05	45.61	35.81	0.819	177.1
FD004	dropout_20	35.47	27.74	0.907	138.4
FD004	combined	46.33	36.14	0.802	185.5

FD001 is comparatively stable: combined corruption raises RMSE from 18.79 to 20.12 and leaves split coverage at 0.970. FD003 point error also changes modestly, although its 108-cycle mean interval is too wide to interpret perfect coverage as strong sharpness. FD002 and FD004 are substantially more sensitive. Five-percent noise raises FD002 RMSE from 27.80 to 39.33 and FD004 from 29.16 to 45.61; split coverage falls from 0.892 to 0.757 and from 0.948 to 0.819. Combined corruption produces similar or larger losses. Random missing cells are less damaging than small continuous perturbations, suggesting that learned regime boundaries and sensor interactions are sensitive to measurement shift.

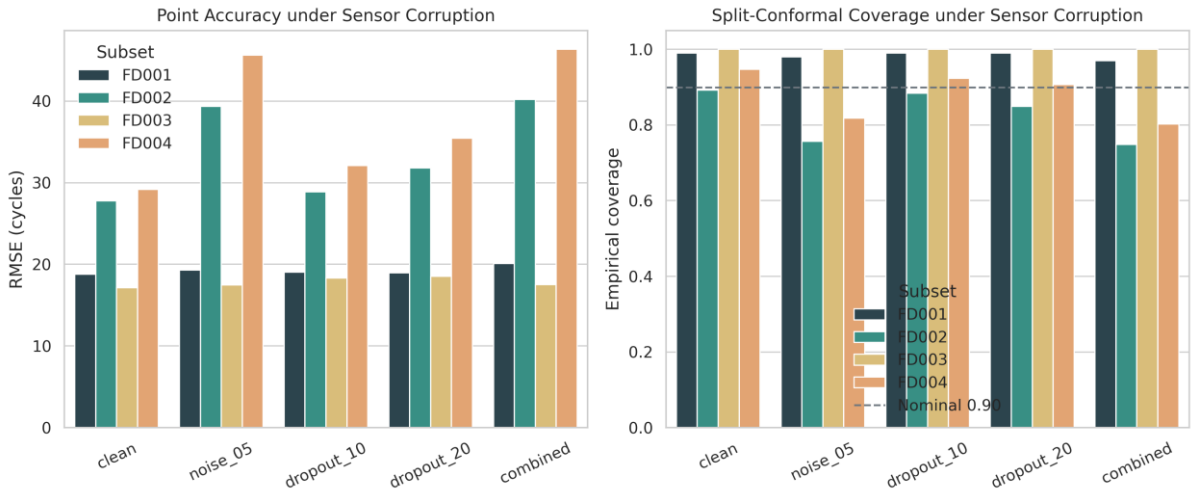

Figure 6. Point RMSE and split-conformal coverage under controlled sensor corruption.

4.6 Descriptive Feature Importance

Permutation on the FD001 official test endpoints identifies cycle (4.45 RMSE increase), sensor 4 (4.43), sensor 11 (3.25), sensor 9 (2.18), and sensor 14 (1.19) as the largest aggregated contributors. The analysis combines each channel's current value, rolling means, and slope. It is descriptive, not causal, and it uses the test set only for interpretation after all models and parameters were fixed; it was not used to select features or tune performance.

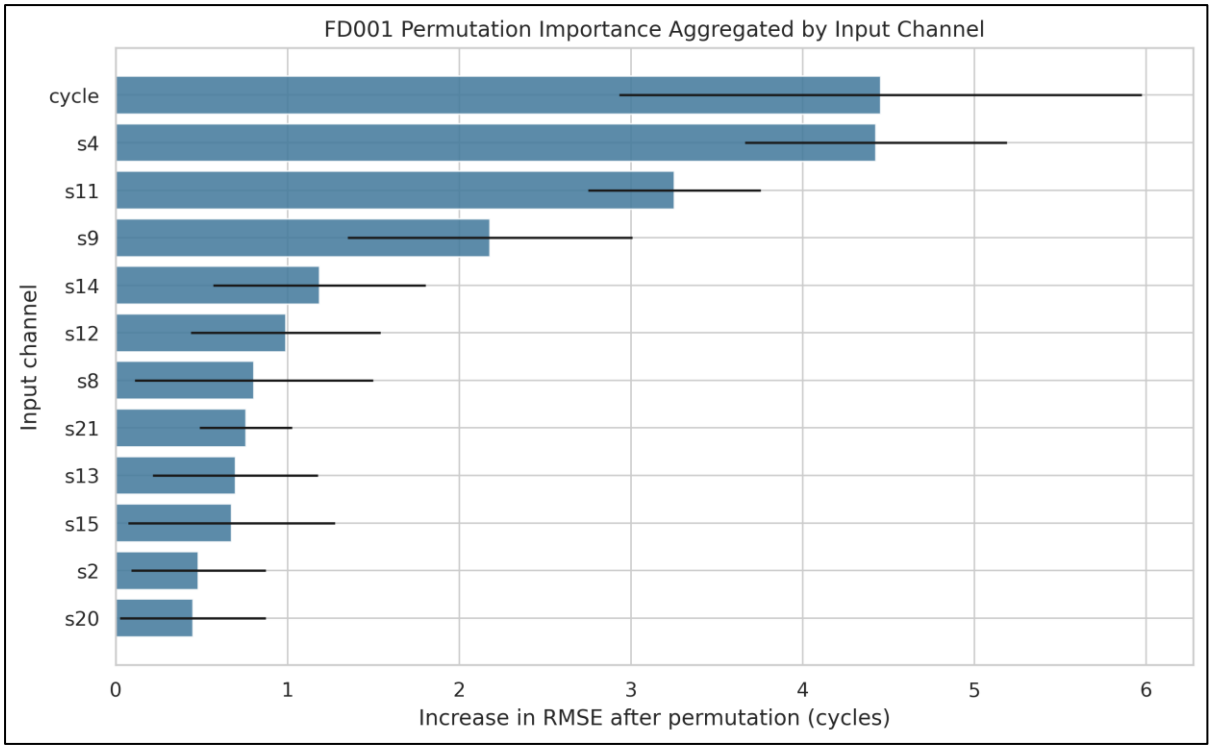

Figure 7. FD001 permutation importance aggregated by raw input channel; bars show mean RMSE increase with variation across repetitions.

5. Discussion

5.1 Answers to the Research Questions

RQ1: Nonlinear ensembles clearly outperform the fleet-life and Ridge baselines, but no single representation wins everywhere. The strongest RMSE is approximately 18–28 cycles depending on subset. Engine-grouped validation and paired tests do not support a universal claim that temporal summaries beat snapshot HGB.

RQ2: Split conformal provides the most dependable marginal coverage in the clean official tests. CQR can be sharper, as on FD001, but fails seriously on FD002 and shows high-RUL undercoverage. Reporting one aggregate coverage number would hide this weakness.

RQ3: Calibration learned on clean pseudo-endpoints does not survive all sensor shifts. Multi-condition FD002 and FD004 are especially sensitive to continuous perturbation. This finding separates in-distribution calibration from deployment robustness.

RQ4: The full transformation and inference path can be packaged in a compact local application. Model artifacts are 0.96–1.07 MB and use 69–97 features. Technical deployability is therefore feasible, but operational authorization is not established by this benchmark.

5.2 Scientific Implications

First, asset identity should be treated as a first-class experimental variable. In predictive-maintenance data, multiple rows are measurements, not independent machines. A leaderboard that does not state its grouping unit is difficult to interpret. Second, uncertainty evaluation should include at least marginal coverage, interval width, and a conditional view over a domain-relevant axis. Third, robustness should be reported separately from calibration. A 90% interval calibrated on clean data cannot be assumed to retain 90% coverage after a sensor process changes.

The result also argues for restrained model claims. Temporal features are intuitively appropriate for degradation, but their benefit depends on operating conditions, faults, target

construction, and the test truncation distribution. The scientifically supported claim is not that one model solves C-MAPSS; it is that a transparent reliability protocol reveals where apparently similar RMSE values conceal different uncertainty and shift behavior.

5.3 Operational Interpretation

A maintenance planner should not treat the interval midpoint as a deterministic failure date. A lower bound may support conservative inspection triage, while interval width signals uncertainty that may justify additional sensing. However, decisions also require cost, criticality, inspection capacity, and an approved safety case. The present prototype deliberately omits prescriptive maintenance commands. It exposes estimates and warnings so that the human decision process remains explicit.

6. Limitations and Threats to Validity

The study has six main limitations. First, C-MAPSS is simulated; its degradation physics and sensor noise do not reproduce the full complexity of an operational fleet. Second, the piecewise target cap at 125 cycles changes the early-life loss surface. Third, one pseudo-endpoint per calibration engine preserves engine-level exchangeability but yields only 25 calibration observations on FD001 and FD003; finite-sample intervals are consequently coarse and conservative. Fourth, the official test truncation mechanism may differ from our uniform 40%–95% pseudo-cutoff mechanism. This mismatch plausibly contributes to CQR failure and prevents an unconditional deployment guarantee.

Fifth, the model family and hyperparameters are deliberately limited; a larger search or sequence model might improve point error but would require nested, engine-grouped selection. Sixth, corruption is synthetic and independent across cells. Real sensor drift is often persistent, correlated, and regime-dependent. We also report unadjusted paired tests as descriptive analyses. These constraints do not invalidate the observed results; they define the conditions under which the conclusions apply.

7. Reproducibility and Practical Use

The accompanying package contains the official NASA archive, the extracted-data checksum, deterministic scripts, environment requirements, trained conformal artifacts for FD001–FD004, all raw CSV result tables, publication figures in PNG and SVG, and the bilingual application. The run manifest records seed 42, Python and library versions, source identity, and every generated artifact. Executing run_experiments.py from the package root regenerates the quantitative outputs; validate_feature_modes.py regenerates the matched snapshot/temporal cross-validation audit.

For a quick application test, run `python app/predict.py --subset FD001 --input app/demo_FD001_unit1.txt`. The bundled example contains 31 observed cycles and returns a point estimate of 122.72 cycles, a split-conformal interval of [75.35, 170.08], and a CQR interval of [101.14, 138.00]. These values demonstrate the executable path; they are not maintenance advice.

8. Conclusion

We completed an engine-disjoint, uncertainty-aware RUL benchmark and a working research application on all four NASA C-MAPSS subsets. Tree ensembles reduce endpoint error substantially relative to linear and fleet-life baselines, but the winning representation changes by subset. Split conformal attains the most consistent clean-test marginal coverage, while CQR exposes strong high-RUL undercoverage on FD002. Sensor noise then shows that clean calibration is not equivalent to robustness. We recommend that future C-MAPSS and industrial

RUL studies report engine-level separation, causal feature timing, calibration sample size, marginal and conditional interval metrics, and covariate-shift stress tests. Future work should test online or weighted conformal calibration, explicit shift detection, regime-aware modeling, and real-flight datasets before any safety-critical use.

Compliance with ethical standards

Disclosure of conflict of interest

The authors declare that they have no conflict of interest.

References

[1] NASA Prognostics Center of Excellence, "Turbofan Engine Degradation Simulation Data Set," NASA Open Data Portal. [Online]. Available: https://data.nasa.gov/dataset/cmapss-jet-engine-simulated-data. Accessed: Aug. 12, 2026.

[2] A. Saxena, K. Goebel, D. Simon, and N. Eklund, "Damage propagation modeling for aircraft engine run-to-failure simulation," in Proc. Int. Conf. Prognostics and Health Management, 2008, pp. 1–9, doi: 10.1109/PHM.2008.4711414.

[3] E. Ramasso and A. Saxena, "Performance benchmarking and analysis of prognostic methods for CMAPSS datasets," Int. J. Prognostics and Health Management, vol. 5, no. 2, 2014, doi: 10.36001/ijphm.2014.v5i2.2236.

[4] F. O. Heimes, "Recurrent neural networks for remaining useful life estimation," in Proc. Int. Conf. Prognostics and Health Management, 2008, pp. 1–6, doi: 10.1109/PHM.2008.4711422.

[5] G. S. 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, LNCS 9642, 2016, pp. 214–228, doi: 10.1007/978-3-319-32025-0_14.

[6] S. Zheng, K. Ristovski, A. Farahat, and C. Gupta, "Long short-term memory network for remaining useful life estimation," in Proc. IEEE Int. Conf. Prognostics and Health Management, 2017, pp. 88–95, doi: 10.1109/ICPHM.2017.7998311.

[7] 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, pp. 1–11, 2018, doi: 10.1016/j.ress.2017.11.021.

[8] A. L. Ellefsen, E. Bjørlykhaug, V. Æsøy, S. Ushakov, and H. Zhang, "Remaining useful life predictions for turbofan engine degradation using semi-supervised deep architecture," Reliability Engineering & System Safety, vol. 183, pp. 240–251, 2019, doi: 10.1016/j.ress.2018.11.027.

[9] F. Wu, Q. Wu, Y. Tan, and X. Xu, "Remaining useful life prediction based on deep learning: A survey," Sensors, vol. 24, no. 11, art. 3454, 2024, doi: 10.3390/s24113454.

[10] D. An, N. H. Kim, and J.-H. Choi, "Practical options for selecting data-driven or physics-based prognostics algorithms with reviews," Reliability Engineering & System Safety, vol. 133, pp. 223–236, 2015, doi: 10.1016/j.ress.2014.09.014.

[11] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery health prognostics: A systematic review from data acquisition to RUL prediction," Mechanical Systems and Signal Processing, vol. 104, pp. 799–834, 2018, doi: 10.1016/j.ymssp.2017.11.016.

[12] A. Javanmardi and E. Hüllermeier, "Conformal prediction intervals for remaining useful lifetime estimation," Int. J. Prognostics and Health Management, vol. 14, no. 2, 2023, doi: 10.36001/ijphm.2023.v14i2.3417.

[13] J. Lei, M. G'Sell, A. Rinaldo, R. J. Tibshirani, and L. Wasserman, "Distribution-free predictive inference for regression," J. American Statistical Association, vol. 113, no. 523, pp. 1094–1111, 2018, doi: 10.1080/01621459.2017.1307116.

[14] Y. Romano, E. Patterson, and E. J. Candès, "Conformalized quantile regression," in Advances in Neural Information Processing Systems, vol. 32, 2019.

[15] A. N. Angelopoulos and S. Bates, "Conformal prediction: A gentle introduction," Foundations and Trends in Machine Learning, vol. 16, no. 4, pp. 494–591, 2023, doi: 10.1561/2200000101.

[16] G. Shafer and V. Vovk, "A tutorial on conformal prediction," J. Machine Learning Research, vol. 9, pp. 371–421, 2008.

[17] R. F. Barber, E. J. Candès, A. Ramdas, and R. J. Tibshirani, "Conformal prediction beyond exchangeability," Annals of Statistics, vol. 51, no. 2, pp. 816–845, 2023, doi: 10.1214/23-AOS2276.

[18] E. Hüllermeier and W. Waegeman, "Aleatoric and epistemic uncertainty in machine learning: An introduction to concepts and methods," Machine Learning, vol. 110, no. 3, pp. 457–506, 2021, doi: 10.1007/s10994-021-05946-3.

[19] L. Biggio, A. Wieland, M. A. Chao, I. Kastanis, and O. Fink, "Uncertainty-aware prognosis via deep Gaussian process," IEEE Access, vol. 9, pp. 123517–123527, 2021, doi: 10.1109/ACCESS.2021.3110049.

[20] K. T. P. Nguyen, K. Medjaher, and C. Gogu, "Probabilistic deep learning methodology for uncertainty quantification of remaining useful lifetime of multi-component systems," Reliability Engineering & System Safety, vol. 222, art. 108383, 2022, doi: 10.1016/j.ress.2022.108383.

[21] L. Basora, A. Viens, M. A. Chao, and X. Olive, "A benchmark on uncertainty quantification for deep learning prognostics," Reliability Engineering & System Safety, vol. 253, art. 110513, 2025, doi: 10.1016/j.ress.2024.110513.

[22] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," Annals of Statistics, vol. 29, no. 5, pp. 1189–1232, 2001, doi: 10.1214/aos/1013203451.

[23] L. Breiman, "Random forests," Machine Learning, vol. 45, pp. 5–32, 2001, doi: 10.1023/A:1010933404324.

[24] P. Geurts, D. Ernst, and L. Wehenkel, "Extremely randomized trees," Machine Learning, vol. 63, no. 1, pp. 3–42, 2006, doi: 10.1007/s10994-006-6226-1.

[25] F. Pedregosa et al., "Scikit-learn: Machine learning in Python," J. Machine Learning Research, vol. 12, pp. 2825–2830, 2011.

[26] D. R. Roberts et al., "Cross-validation strategies for data with temporal, spatial, hierarchical, or phylogenetic structure," Ecography, vol. 40, no. 8, pp. 913–929, 2017, doi: 10.1111/ecog.02881.

[27] T. Gneiting and A. E. Raftery, "Strictly proper scoring rules, prediction, and estimation," J. American Statistical Association, vol. 102, no. 477, pp. 359–378, 2007, doi: 10.1198/016214506000001437.

[28] A. Saxena, J. Celaya, B. Saha, S. Saha, and K. Goebel, "Metrics for evaluating performance of prognostic techniques," in Proc. Int. Conf. Prognostics and Health Management, 2008, pp. 1–17, doi: 10.1109/PHM.2008.4711436.

[29] F. Wilcoxon, "Individual comparisons by ranking methods," Biometrics Bulletin, vol. 1, no. 6, pp. 80–83, 1945, doi: 10.2307/3001968.

