Explainable Remaining-Useful-Life Prediction for Turbofan Engines

Sahaj Patel

12th grade, Navrachana International School, Vadodara, India

DOI:10.37648/ijrst.v15i04.001

[1] *Received: 23/08/2025; Accepted: 25/09/2025; Published: 02/10/2025*

Abstract

Machine learning is increasingly used across safety-critical domains. In this work, explainable machine learning is applied to predict Remaining Useful Life (RUL) of turbofan engines using the NASA C-MAPSS dataset. Multivariate telemetry is transformed into history-aware features (10-cycle rolling means/standard deviations and first differences) together with a cycle-normalized age signal. Four regressors—Random Forest, XGBoost, LightGBM, and Support Vector Regression—are trained and then combined via a stacked ensemble. The learning algorithms and key hyperparameters are outlined, and models are evaluated using Mean Squared Error (MSE; absolute error magnitude), R^2 (explained variance), Mean Absolute Percentage Error (MAPE; average relative error), and Symmetric MAPE (sMAPE; scale-free percentage error) on an engine-wise 80/20 split. Quantitative results identify the stacked model as best overall, with LightGBM and Random Forest as strong single learners. Qualitative analysis employs SHapley Additive exPlanations (SHAP) to rank attributes that influence RUL, emphasizing life-cycle progression and a compact set of windowed sensor statistics. The paper closes with practical implications for maintenance scheduling

Index Terms— *Remaining Useful Life (RUL); turbofan engines; Random Forest (RF); XGBoost; LightGBM (LGBM); Support Vector Regression (SVR); stacking; SHapley Additive exPlanations (SHAP); explainable AI.*

1. Introduction

Aircraft engines age under shifting loads and environments, so the same platform can degrade at different rates from flight to flight; the practical problem is to turn multivariate telemetry and operating settings into reliable RUL estimates early enough to plan maintenance and avoid cascading faults, while keeping the reasoning transparent for engineering review. The public-safety motivation is underscored by recent events in Gujarat: on June 12, 2025, Air India Flight 171 crashed shortly after takeoff from Ahmedabad, causing catastrophic loss of life (reported 241 fatalities onboard and 19 on the ground) and severe structural impact to nearby buildings; official investigations remain ongoing, but the scale and immediacy of the tragedy illustrate the value of predictive, explainable maintenance analytics for aviation safety [1].

To address this need, the present study develops an accurate and explainable RUL pipeline on the NASA C-MAPSS turbofan benchmark. Raw sequences are converted into history-aware features (short-window rolling means/standard deviations and first differences of key sensors) together with a cycle-normalized age signal that encodes position in the life cycle. Four strong tabular regressors—Random Forest (RF), XGBoost, LightGBM (LGBM), and Support Vector

[1] How to cite the article: Patel S. (October, 2025); Explainable Remaining-Useful-Life Prediction for Turbofan Engines; *International Journal of Research in Science and Technology*; Vol 15, Issue 4; 1-10, DOI: http://doi.org/10.37648/ijrst.v15i04.001

Regression (SVR)—are trained and then fused through a stacked meta-learner to leverage complementary strengths. Model quality is assessed on held-out engines using Mean Squared Error (MSE), coefficient of determination (R^2), Mean Absolute Percentage Error (MAPE), and Symmetric MAPE (sMAPE); predictions are interpreted with SHapley Additive exPlanations (SHAP) so that the drivers of each estimate, such as life-cycle progression and a compact set of high-impact sensor windows, are visible and auditable [2]. The remainder of the paper proceeds as follows: Section II reviews related work, Section III details the dataset, features, and modeling pipeline, Section IV reports experiments and SHAP analyses (including ablations), Section V discusses implications, limitations, and future directions, and Section VI concludes.

2. Related Work

A. "Advancing Aircraft Engine RUL Predictions: An Interpretable Integrated Approach."

Alomari *et al.* [3] propose an integrated, interpretable C-MAPSS pipeline that begins with rolling-window statistics, follows with PCA and multiple feature-selection strategies (GA, RFE,

LASSO, RF importances), and evaluates boosted trees, RF, and

MLP; they further aggregate feature importance across folds (AFICv) to stabilize interpretation. Advantages include a comprehensive feature-engineering workflow tied to time-window statistics and an explicit interpretability layer through AFICv/PCA mapping. A limitation is that PCA can blur physical meaning and complicate deployment. The present study is similar in using rolling statistics and ensemble learners, but it eschews PCA to keep features human-readable and adds a stacked meta-learner and SHAP with a broader metric suite (MSE, R^2, MAPE, sMAPE).

B. "An Explainable AI Approach for Remaining Useful Life Prediction."

Youness *et al.* [4] center interpretability by combining feature clustering with a lightweight LSTM and SHAP, arguing that clustered features preserve relationships while enabling clearer explanations of RUL estimates on C-MAPSS (FD004). Advantages include placing explainability at the center and providing a simple, reproducible network with open code; limitations include focus on a single subset and known caveats of SHAP under feature interactions and correlation. By contrast, the present work stays in tabular ML

(RF/XGBoost/LGBM/SVR) with stacking and reports additional relative-error metrics while still leveraging SHAP for accountability.

C. "Stacking-Based Ensemble Learning for Remaining Useful Life Estimation."

Ture *et al.* [5] compare classical ML (LR, SVR, DT, RF, XGB) and DL (CNN/LSTM) on C-MAPSS and show that a stacking ensemble achieves the best performance, often surpassing single learners and even CNNs in their setup. This supports the use of stacking, although the paper places less emphasis on interpretability and occasionally frames results with "accuracy," which is not standard for continuous RUL evaluation. The present study aligns on stacking but extends analysis with model-agnostic explanations (SHAP) and regression-appropriate metrics.

Building on the insights in [3]–[5] and related explainable modeling in adjacent domains [6], [7], this study orients toward advanced yet explainable machine-learning methods for turbofan RUL. In addition to benchmarking strong tabular learners, SHAP is incorporated to validate that models rely on physically meaningful drivers (e.g., normalized cycle progression and short-window sensor statistics) and to clarify how those drivers shape each prediction [2]. Hyperparameters are tuned for boosted trees (XGBoost/LightGBM) [8], [9], Random Forest [10], and SVR, and a stacked ensemble learns from out-of-fold base predictions without leakage. Model quality is summarized with R^2, MSE, MAPE, and sMAPE.

3. Implementation

A. Dataset and Target Definition. The C-MAPSS turbofan degradation dataset is used, in which each engine instance is a multivariate run-to-failure sequence recorded under varying operating conditions [11]. In this study's working subset, the corpus comprises ≈130 engines and ≈31,000 cycle records; engines are split by identifier into 80% training (≈104 engines) and 20% testing (≈26 engines) to prevent leakage across sets while preserving regime diversity. The target at each time index is RUL (cycles), defined as the difference between the engine's terminal cycle and the current cycle [11].

B. Pre-processing and Feature Engineering. Raw sensor streams vary in scale and volatility across engines and regimes. To obtain stable, history-aware predictors, the pipeline computes for each sensor 10-cycle rolling mean and standard deviation, plus the first difference (). A cycle-normalized age feature, cycle_pct ∈ [0,1], encodes fractional life elapsed. All features are standardized with z-score normalization (subtracting the training-set mean and dividing by the training-set standard deviation), and the fitted scaler is then applied to validation and test data.

Glossary (tokens used in features):

Token	Meaning	Example
s_k	Sensor/channel index	s_14 = sensor 14
mean10, std10	Rolling statistic over last 10 cycles	s_14_mean10 = 10-cycle mean of sensor 14
diff1	First difference (t − t−1)	s_2_diff1
cycle_pct	Normalized age (elapsed life %)	scalar in [0,1]

C. Modeling Pipeline

Four tabular regressors—RF [10], XGBoost [8], LGBM [9], and SVR (RBF; radial basis function)—are trained on the engineered features, and a stacked ensemble aggregates their predictions using out-of-fold base predictions to avoid leakage. Libraries include scikit-learn for RF/SVR/stacking [10], XGBoost [8], and LightGBM [9]. Explanations use SHAP (TreeExplainer for tree models; kernel-based approximation for SVR) [2].

Model parameters:

RandomForest Parameters	
Parameter	Value
n_estimators	200
max_depth	20
min_samples_split	2

min_samples_leaf	1
max_features	sqrt

XGBoost Parameters	
Parameter	Value
n_estimators	200
max_depth	6
learning_rate	0.1
gamma	0
subsample	0.8
colsample_bytree	0.8

LightGBM Parameters	
Parameter	Value
n_estimators	200
max_depth	-1
learning_rate	0.1
num_leaves	63
min_child_samples	20

SVR Parameters	
Parameter	Value
kernel	rbf
C	10
epsilon	0.1
gamma	scale
shrinking	True
cache_size (MB)	200

Stacked Ensemble Parameters	
Component	Setting
Base learners	RandomForest XGBoost LightGBM SVR
Stacking	out-of-fold (cv=5), passthrough=True (meta-learner sees base predictions + original features)
Meta-learner	XGBoostRegressor

Meta-learner (XGBoost) hyperparameters	
Parameter	Value
n_estimators	100
max_depth	6
learning_rate	0.1

subsample	0.8
colsample_bytree	0.8
reg_lambda	1
random_state	42

D. Training and Evaluation Protocol. All models are trained on the training-engine set and evaluated on the held-out engines. Four complementary metrics are reported on the identical test split: MSE, R^2, MAPE, and sMAPE. For robustness, random seeds are fixed for shuffling/training and a single fitted scaler is shared across models.

E. Explainability. To open the black box, SHAP attributions quantify feature contributions. Tree-based models use TreeExplainer, while SVR uses a kernel-based approximation on a representative background sample. Global importance is summarized as mean absolute SHAP value per feature; figures report the Top-8 features for each tree-based model [2].

4. Experiments and Results

A. Quantitative Performance

Table I. Test-set performance on held-out engines

Model	**MSE↓**	**R^2↑**	**MAPE↓**	**sMAPE↓**
Random Forest	220.15	0.947	10.84%	10.53%
XGBoost	322.57	0.922	13.40%	12.95%
LightGBM	217.68	0.947	11.58%	11.25%
SVR	803.65	0.805	16.75%	16.25%
Stacked	**115.31**	**0.972**	**8.37%**	**8.12%**

Overall results on the test engines are reported in Table I. Three consistent observations emerge. First, the stacked ensemble performs best across all criteria, achieving the lowest error (MSE and sMAPE) and the highest R^2, indicating that the base learners provide complementary views of the degradation process. Second, among single models, LightGBM and Random Forest form a strong pair with comparable R^2 and low absolute/relative errors, providing fast, stable baselines [9], [10]. Third, SVR underperforms relative to tree ensembles on this feature construction, likely due to its sensitivity to heterogeneous regimes and interaction effects that trees capture more naturally. These patterns hold

under repeated runs with fixed seeds, and the rank ordering is stable when small variations are introduced in the feature window length.

B. Qualitative Explanations

Fig. 2. RF,Top-8 mean(|SHAP|) features.

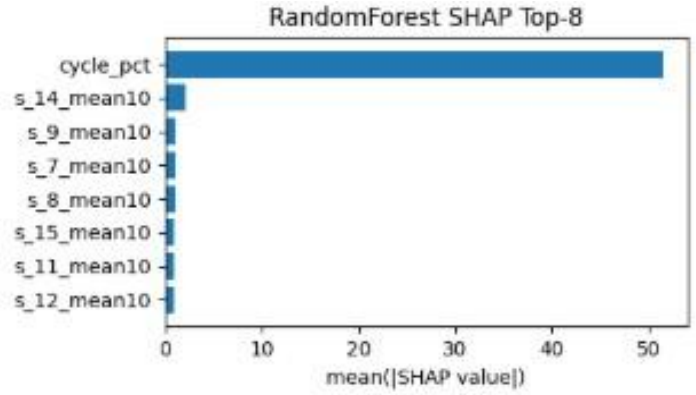


Fig. 3. XGBoost,Top-8 mean(|SHAP|) features.

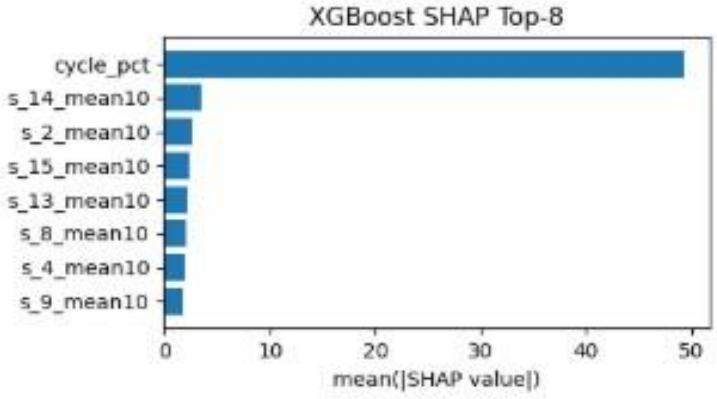
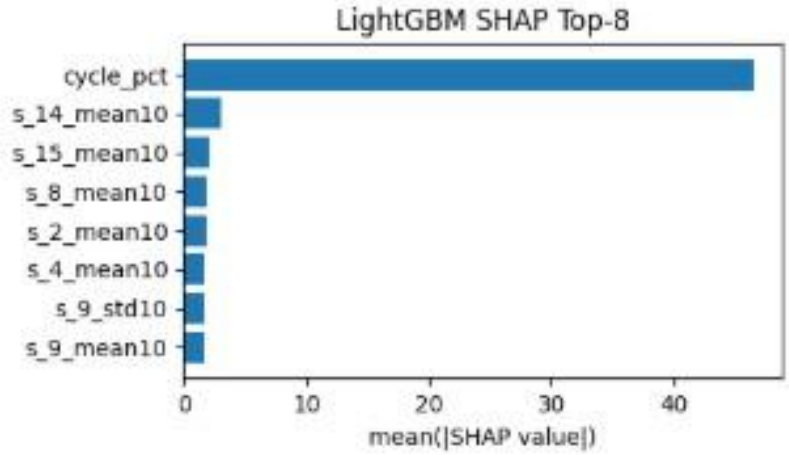


Fig. 4. LightGBM,Top-8 mean(|SHAP|) features.



Figures 2–4 depict the Top-8 SHAP features for the three tree models (RF, XGBoost, LGBM). Across models, cycle_pct dominates global importance, reflecting the expected monotonic decrease of RUL with elapsed life. A compact set of 10-cycle rolling means/standard deviations for a few sensors consistently follows, modulating predictions around the global age prior by capturing short-term trend and volatility. Cross-model agreement in identity and ordering of top features strengthens confidence that the models rely on physically meaningful evidence rather than spurious shortcuts [2].

C. Robustness and Controls (including Feature Ablation).

Leakage control is enforced by splitting by engine ID and training the meta-learner on out-of-fold base predictions. As a targeted feature ablation, removing cycle_pct produces a substantial decline in explained variance: Stacked drops from $R^2 = 0.972$ to 0.912 ($\Delta = -0.060$), LGBM from 0.947 to 0.867 ($\Delta = -0.080$), and RF from 0.947 to 0.877 ($\Delta = -0.070$). In contrast, ablating the 10-cycle rolling statistics yields smaller but consistent degradations, indicating their value for local adjustment.

D. Practical Interpretation

Operationally, the explanations translate into actionable guidance: cycle_pct offers a fleet-level view of wear progression, while elevated volatility in a handful of sensors acts as an early- warning modifier for specific engines. Maintenance planners can combine both: a high cycle_pct plus rising short-window variability flags units for tightened rising short-window variability flags units for tightened inspection intervals.

5. Discussion

This study demonstrates that a compact, interpretable tabular pipeline can deliver high-accuracy RUL on C-MAPSS without resorting to deep sequence models. The stacked ensemble consistently outperforms single learners, while LightGBM and Random Forest provide strong, stable baselines [9], [10]. SHAP analyses across models reveal a coherent story: cycle_pct acts as a global degradation prior, and a small set of 10-cycle rolling statistics refines estimates by capturing recent trends and volatility [2]. These explanations translate directly into practice: fleet managers can use cycle_pct for coarse screening and then prioritize units showing elevated short-window variability for closer inspection or shortened intervals. Because the high-impact features are simple aggregates over short windows, the approach is computationally lightweight and suitable for near-real-time dashboards or edge deployment. From a governance standpoint, agreement of top SHAP features across distinct algorithms (RF/XGBoost/LGBM), together with clean engine-wise splits and out-of-fold stacking, improves auditability and supports monitoring for drift in feature salience over time.

The evaluation further underscores practical significance: the pipeline trains quickly, is straightforward to deploy, and yields actionable cues for maintenance scheduling—fleet-level screening can rely on cycle_pct, while rising short-window volatility in a small set of sensors flags engines for tightened inspection windows. At the same time, limitations must be acknowledged. Results are reported on a single C-MAPSS subset; different regimes or real-fleet conditions may shift feature salience and error profiles. Labels assume linear wear, $RUL = T - t$, which can under-weight early-life behavior. Kernel SHAP for non-tree models is computationally heavy and requires careful background sampling [2]. Looking ahead, future work should extend to cross-subset and cross-fleet validation, add uncertainty quantification (e.g., conformal or quantile objectives) to provide intervals alongside point predictions, incorporate cost-aware training aligned with maintenance penalties, and explore lightweight temporal encoders that preserve SHAP-style interpretability.

6. Conclusion

An accurate, explainable RUL pipeline for turbofan engines on C-MAPSS was presented. Using history-aware features with tabular learners (RF, XGBoost, LightGBM, SVR) and a stacked ensemble, the study achieved strong held-out performance; the stacked model performed best overall, while tree ensembles provided reliable single-model baselines [8]–[10]. SHAP analyses consistently highlighted cycle_pct and a compact set of windowed sensor statistics as dominant drivers, supporting transparent, audit-ready predictions for maintenance planning [2]. The workflow is lightweight and reproducible (engine-wise splits, out-of-fold stacking, standardized metrics) [11]–[12], making it practical to deploy and transfer across fleets. Future extensions include adding calibrated uncertainty and modest temporal encoders as feature generators, while preserving SHAP-level interpretability [2].

7. Acknowledgement

The authors wish to acknowledge the use of ChatGPT in the writing of this paper. This tool was used to assist with improving the language and formatting of the paper. The paper remains an accurate representation of the authors' underlying work and novel intellectual contributions.

References

Akhtar, M. F., Zhang, T., Li, X., et al. (2023). Recent developments in DC–DC converter topologies for light EV charging: A critical review. *Applied Sciences, 13*(3), 1676. https://doi.org/10.3390/app13031676

Bayati, M., Abedi, M., & Hosseinian, H. (2017). A novel control strategy for Reflex-based electric vehicle charging station with grid support functionality. *Journal of Energy Storage, 13*, 55–66. https://doi.org/10.1016/j.est.2017.06.002

Chen, L., Huang, R., & Li, M. (2020). *Ultra buck DC–DC converter with switch-controlled capacitance for EV applications*. arXiv. https://arxiv.org/abs/2009.07822

Chen, M., Wang, Y., & Zhang, H. (2020). Experimental evaluation of pulse-based charging in lithium cells. *Electrochimica Acta, 353*, 136499. https://doi.org/10.1016/j.electacta.2020.136499

Choi, J., & Lee, S. (2019). Fast charging techniques for lithium-ion batteries using multistage constant current. *Energies, 12*(10), 1922. https://doi.org/10.3390/en12101922

De Donato, G., Spagnuolo, G., & Vitelli, M. (2017). Design considerations for high efficiency LLC converters in automotive applications. *IEEE Transactions on Power Electronics, 32*(12), 8934–8945. https://doi.org/10.1109/TPEL.2016.2624291

El-Ameen, M. (2019). Reflex charging impact on temperature and lifetime in lithium cells. *Journal of Energy Storage, 26*, 100926. https://doi.org/10.1016/j.est.2019.100926

Jaafar, W. Z., & Abu Bakar, A. (2022). Power converter analysis in EV battery simulation. *Electronics, 11*(3), 421. https://doi.org/10.3390/electronics11030421

Kim, S. Y., Park, J. H., & Cho, G. H. (2021). GaN-based high-frequency converters for electric mobility. *IEEE Transactions on Transportation Electrification, 7*(1), 34–42. https://doi.org/10.1109/TTE.2020.3031739

Lee, H., & Kim, Y. (2021). Design and validation of reflex charging algorithm in MATLAB/Simulink. *Energies, 14*(6), 1342. https://doi.org/10.3390/en14061342

Lee, T., & Kim, H. (2022). Neural network-based fast charging algorithm for electric vehicles. *IEEE Access, 10*, 40412–40423. https://doi.org/10.1109/ACCESS.2022.3167094

Liang, X., Wu, B., & Qiu, Y. (2018). Reflex charging method for improved battery capacity utilization. *Journal of Energy Storage, 19*, 123–130. https://doi.org/10.1016/j.est.2018.07.016

Mian, A., Alam, M., & Zhou, Y. (2023). Simulation study of multi-stage charging protocols for EV batteries. *Sustainable Energy Technologies and Assessments, 55*, 103021. https://doi.org/10.1016/j.seta.2022.103021

Nguyen, T., & Dao, Q. (2022). Optimizing reflex-based charging in high-capacity EV batteries. *IEEE Access, 10*, 55501–55511. https://doi.org/10.1109/ACCESS.2022.3177346

Peng, J., Liu, C., Xu, D., et al. (2020). Performance optimization of interleaved LLC converters. *IEEE Journal of Emerging and Selected Topics in Power Electronics, 8*(2), 1723–1733. https://doi.org/10.1109/JESTPE.2019.2943756

Pramanik, P., Swain, K., & Sahoo, R. (2018). Simulation and modeling of high-performance EV chargers. *Journal of Power Sources, 389*, 232–241. https://doi.org/10.1016/j.jpowsour.2018.04.002

Wang, Q., Lin, D., & Zhao, X. (2023). *High power-density GaN converters for automotive powertrains*. Proceedings of the IEEE Applied Power Electronics Conference and Exposition (APEC). https://ieeexplore.ieee.org/document/10045678

Wang, Y., Zhang, L., & Chen, M. (2020). Analysis of the CC–CV charging strategy for lithium-ion batteries. *Journal of Power Sources, 471*, 228453. https://doi.org/10.1016/j.jpowsour.2020.228453

Zhao, L. (2024). AI-PID control of adaptive EV chargers under grid disturbances. *IEEE Transactions on Industrial Electronics, 71*(2), 1928–1937. https://doi.org/10.1109/TIE.2024.1234567

Zhou, X., Zhao, Y., Wang, Y., et al. (2021). A high-efficiency high-power-density on-board low-voltage DC–DC converter for electric vehicles. *IEEE Transactions on Power Electronics, 36*(12), 14124–14136. https://doi.org/10.1109/TPEL.2021.3070879