International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 13 Issue: 10 | Oct 2026 www.irjet.net p-ISSN: 2395-0072

Investigation and Implementation of Deep Learning Architectures for Telematics-Based Component Degradation Analysis and Remaining Useful Life Estimation in Two-Wheeler Applications

Omkar Dhembre[1], Prateek Salian [2], Prasad Pednekar[3], Ashwini Kadam [4]

[123]Masters of Computer Applications, School of Humanities and Sciences, D. Y Patil Deemed to be University, India.
[4]Associate Professor, School of Humanities and Sciences, D. Y Patil Deemed to be University, India.

Abstract - *Brake wear is a safety-critical maintenance concern for two-wheelers, while fixed-interval servicing may not reflect the actual condition of a component. This study develops a data-driven regression model to estimate brake remaining useful life (RUL), expressed in kilometres, from telemetry and usage indicators. The supplied notebook uses 3,000 records from a workbook named two_wheeler_simulated_brake_rul_dataset.xlsx. Seven input features speed, distance, vibration, temperature, acceleration, brake usage, and component age are standardized and used to train a feed-forward artificial neural network (ANN) with hidden layers of 64, 32, and 16 neurons, ReLU activations, dropout, and a linear output. An 80:20 train–test split provides 2,400 training records and 600 test records; early stopping monitors validation loss. On the saved test output, the model obtains a mean absolute error (MAE) of 1,175.77 km, root mean squared error (RMSE) of 1,457.99 km, and coefficient of determination (R^2) of 0.7912. The notebook also includes training-history, actual-versus-predicted, and residual plots. These results indicate that the model captures a substantial portion of target variation, but the average error remains operationally important for maintenance decisions. Because the dataset is simulated and no baseline comparison, independent real-vehicle validation, or inferential hypothesis test is recorded, the model should be treated as a proof of concept. Future work should validate on real brake-wear measurements, compare alternative algorithms, quantify uncertainty, and evaluate safety-oriented decision thresholds.*

Keywords - Remaining useful life (RUL); predictive maintenance; artificial neural network; brake wear; vehicle telemetry; regression.

INTRODUCTION:

Remaining useful life estimation predicts the operating time or usage remaining before a component reaches a defined end-of-life condition. In vehicles, brake components gradually wear under braking events, temperature changes, loading, road conditions, and driving style. A reliable estimate can support condition-based maintenance, reduce unnecessary inspections, and help avoid unexpected degradation. However, brake condition is safety-critical: a model output must not replace required inspections or manufacturer service limits.

Data-driven RUL approaches learn a relationship between sensor or operating features and a target lifetime label. Recent reviews describe a broad range of statistical, machine-learning, deep-learning, and physics-informed approaches, while also noting challenges around data quality, generalization, interpretability, and practical validation [1]–[4]. This study implements a compact ANN regressor using seven features available in the supplied simulated telemetry dataset.

PROBLEM STATEMENT:

The technical problem is to estimate Brake_RUL_km from a fixed set of seven usage and telemetry variables. The research asks whether a feed-forward ANN trained on the supplied simulated dataset can produce useful continuous RUL estimates on a held-out test subset. The main practical concern is the size and distribution of prediction errors, because an error of several hundred or thousand kilometres may influence service timing.

LITERATURE REVIEW:

Research on RUL prediction has evolved from statistical and physics-based approaches toward machine learning and deep neural networks. The following recent publications provide context. Their results are not directly comparable with this notebook because they use different assets, datasets, targets, and evaluation protocols.

Study	Method / data context	Reported contribution	Limitations relevant to this study
Wu et al. (2024) [1]	Survey of deep-learning RUL methods across equipment domains.	Organizes deep-learning approaches and discusses challenges and future directions.	A survey, not a direct benchmark for two-wheeler brakes; cross-domain findings require task-specific validation.
Li et al. (2024) [2]	Review of physics-informed, data-driven RUL prediction.	Discusses combining physical knowledge with data-driven models to improve prognosis.	Physics-informed methods need trustworthy physical models and suitable measurements.
Qiao et al. (2024) [3]	Critical review of ML-based RUL methods, especially turbofan engines and bearings.	Examines data preprocessing, estimation methods, and recurring limitations.	Asset-specific benchmarks differ from brake wear and cannot establish expected performance here.
Iqbal, Bhambri & Lahase (2024) [4]	SAE technical paper on brake-pad life monitoring; hybrid approach with vehicle dynamics and deep neural networks; rig data with limited vehicle validation.	Reports 94.8% validation accuracy for its particular framework and motivates indirect brake-life monitoring.	Metric and experimental design differ from MAE/RMSE/R^2 here; limited vehicle-level data was noted.
Brake-pad wear study (2024) [5]	On-road passenger-car brake data; accounts for varying vehicle mass and estimates friction-brake work.	Uses 10,200 km of on-road data and 31 brake-wear measurement samples.	Passenger cars and a different measurement setup; sample count for direct wear measurements is limited.
Wang et al. (2025) [6]	Survey of graph neural networks for RUL prediction.	Reviews graph construction, graph modelling, information processing, readout, and evaluation.	GNNs require suitable relational structure and may add complexity not justified by a small tabular dataset.
Tsallis et al. (2025) [7]	Systematic review of ML-based predictive maintenance across application domains.	Compares datasets, techniques, metrics, and implementation challenges; highlights data heterogeneity and deployment issues.	Broad industrial review rather than a two-wheeler brake experiment.
Pan et al. (2025) [8]	Review of deep-learning RUL prediction for equipment components and sustainable manufacturing.	Summarizes deep-learning approaches and practical challenges for component RUL.	Findings are broad; real-world deployment still depends on representative data and validation.

Synthesis:

Recent work supports data-driven RUL estimation, but stresses that dataset choice, operating conditions, sensor quality, uncertainty, and validation design strongly affect the reliability of reported results. Brake-specific research is particularly relevant [4], [5]. The present notebook is narrower: it tests a tabular ANN on simulated two-wheeler telemetry and reports regression metrics.

Research Gap:

The supplied experiment addresses a useful prototype problem, but several gaps remain. First, the notebook uses simulated data, so the learned relationship may not represent real brake-pad wear. Second, it evaluates one ANN architecture without comparing it against linear regression, random forest, gradient boosting, or sequence-based models. Third, no confidence or prediction intervals are produced. Fourth, the split is random rather than grouped by vehicle, rider, trip, or time; correlated records could make generalization appear stronger than it is. Finally, the saved notebook does not report a formal statistical significance test or an operational cost/safety evaluation.

Research Questions:

RQ1: How accurately does the ANN estimate brake RUL on the held-out test subset, as measured by MAE, RMSE, and R^2?

RQ2: How closely do predicted RUL values follow the actual values, and what patterns are visible in the residual plot?

RQ3: Does prediction error vary across actual RUL ranges?

RQ4: Can the current evidence justify real-world maintenance use, or is additional validation required?

Research Objectives:

1. To prepare the dataset by removing duplicate rows, handling invalid values, and excluding missing required values and negative target RUL.

2. To train an ANN regressor using seven standardized input features and an 80:20 train–test split.

3. To evaluate the model using MAE, RMSE, and R^2 on 600 held-out records.

4. To interpret the saved training, prediction, and residual graphs and inspect errors across RUL ranges.

Hypothesis:

H0 (null hypothesis): The ANN does not provide a practically meaningful reduction in prediction error compared with an appropriate baseline regressor on the same held-out data.

H1 (alternative hypothesis): The ANN provides a practically meaningful reduction in prediction error compared with an appropriate baseline regressor on the same held-out data.
Statistical status: this hypothesis is not tested by the supplied notebook because it does not train a baseline model or save the per-record baseline and ANN predictions needed for a paired comparison. Therefore, no p-value or statistical significance claim is made. A suitable future test is a paired bootstrap confidence interval for the difference in MAE (baseline MAE minus ANN MAE), with a pre-defined practical improvement threshold. If testing multiple models, use identical splits and correct for multiple comparisons.

RESEARCH METHODOLOGY:

Research design

Quantitative experimental regression study using a supplied simulated tabular dataset. The notebook fixes NumPy and TensorFlow random seeds to 42.

Dataset and variables

The workbook is named two_wheeler_simulated_brake_rul_dataset.xlsx and the selected worksheet is Telemetry_Data. The notebook reports 3,000 rows and 10 columns. The target is Brake_RUL_km; Trip_ID and Brake_Wear_Level are present in the workbook but are not used as model inputs.

Role	Variable	Description / unit
Input	Speed_kmh	Vehicle speed (km/h)
Input	Distance_km	Distance-related telemetry (km)
Input	Vibration_mm_s	Vibration measurement (mm/s)
Input	Temperature_C	Temperature (°C)
Input	Acceleration_m_s2	Acceleration (m/s^2)
Input	Brake_Usage_per_100km	Brake usage rate per 100 km
Input	Component_Age_km	Component age / accumulated use (km)
Target	Brake_RUL_km	Remaining useful life in kilometres

Preprocessing and split

The code removes duplicate records, replaces positive/negative infinity with missing values, drops rows missing any required input or target, and filters out negative target RUL. It then applies an 80:20 random split with random_state=42. StandardScaler is fitted on the training features and then applied to training and test features. The training subset contains 2,400 records and the test subset contains 600.

ANN architecture and training

Layer	Configuration
Input	7 numeric features
Dense 1	64 neurons, ReLU
Dropout	0.15 dropout rate
Dense 2	32 neurons, ReLU
Dense 3	16 neurons, ReLU
Output	1 neuron, linear activation

Optimizer / loss	Adam (learning rate 0.001); mean squared error
Training controls	Batch size 32; up to 150 epochs; validation_split 0.20; EarlyStopping patience 15; restore_best_weigh

Evaluation procedure

The notebook evaluates test predictions using MAE, RMSE, and R^2. It plots training and validation loss, actual versus predicted RUL, and prediction residual versus actual RUL. It also sorts the test records by absolute error and groups errors into four actual-RUL bands.

Tools and Technologies

Tool / library	Purpose
Python	Programming and experiment orchestration
Jupyter Notebook / Google Colab-compatible upload logic	Interactive execution and saved outputs
Pandas, NumPy	Tabular data preparation and numerical operations
Matplotlib	Training and error visualizations
scikit-learn	Train/test split, standardization, and regression metrics
TensorFlow / Keras	ANN definition, training, early stopping, serialize

Data Analysis and Interpretation:

Dataset and split summary

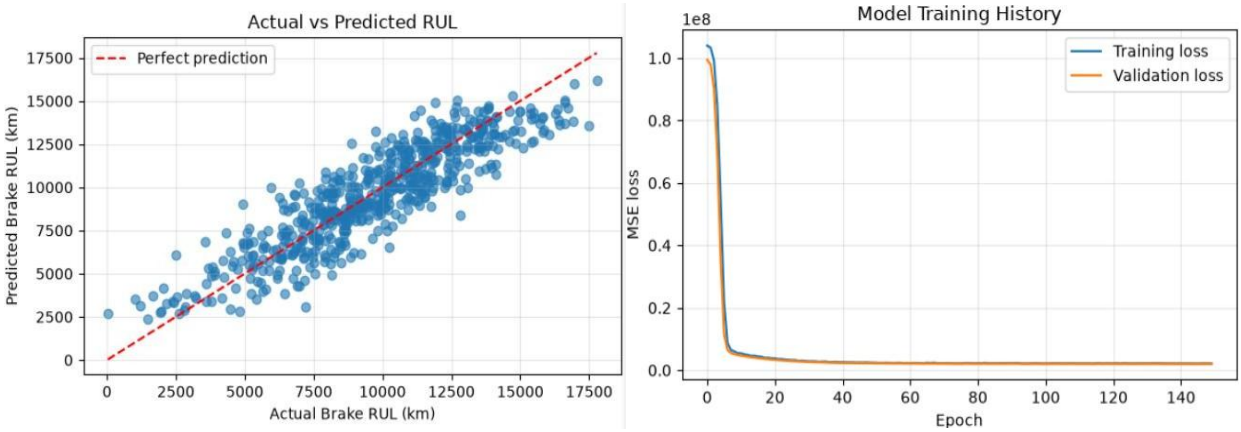
Measure	Notebook result
Original dataset	3,000 rows × 10 columns
Cleaned dataset	3,000 rows × 10 columns
Missing required values after cleaning	0 in each required feature/target column
Training subset	2,400 records
Test subset	600 records
Input features	7
Target mean	9,608.99 km
Target standard deviation	3,312.43 km
Target standard deviation	3,312.43 km
Target range	0–18,000 km

Test performance:

Metric	Saved value	Interpretation
MAE	1,175.77 km	Average absolute prediction error.
RMSE	1,457.99 km	Penalizes large errors more than MAE.
R^2	0.7912	Model explains about 79.12% of target variance on this test split.

The metrics suggest useful predictive signal in the simulated dataset, but an MAE of about 1,176 km is not negligible. Whether this is acceptable depends on brake inspection intervals, failure risk, the intended warning horizon, and the consequences of underestimating remaining life. R^2 alone is not sufficient to establish safety or deployment readiness.

Actual notebook output: training history and actual-versus-predicted plot



Example prediction:

Output	Value
Actual Brake RUL	7,014.84 km
Predicted Brake RUL	6,339.58 km
Signed error (predicted − actual)	−675.26 km
Absolute error	675.26 km

For this displayed sample, the model underestimates RUL by approximately 675 km. This single case illustrates the output format and is not sufficient to characterize overall performance.

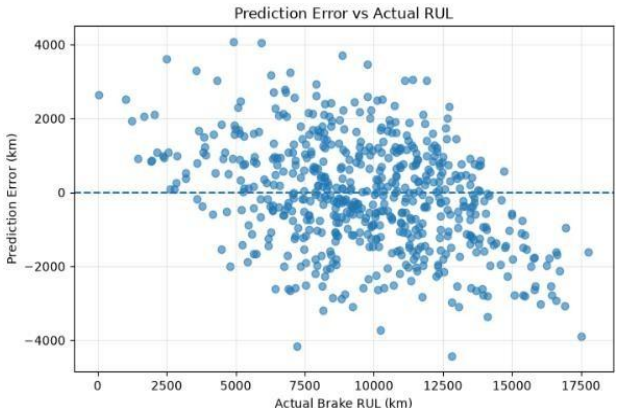
Error distribution and RUL bands:

Absolute error statistic	Value (km)
Count	600
Mean	1,175.77
Standard deviation	862.87
Minimum	3.23

25th percentile	490.14
Median	1,012.25
75th percentile	1,699.11
Maximum	4,418.35

The maximum absolute error of 4,418.35 km shows that a subset of predictions can be substantially wrong. The median absolute error (1,012.25 km) is lower than the mean, consistent with a right tail of larger errors. The notebook also creates a table of errors by actual-RUL bands; exact cell values should be checked against the notebook's displayed table before making band-specific claims.

Actual notebook output: residual plot:



Residuals scattered around zero without a strong pattern would support a better fit; systematic curvature or changing spread could indicate bias or heteroscedasticity. The plot should be reviewed at full resolution before drawing a definitive claim about these patterns.

Hypothesis testing:

No valid inferential hypothesis test can be calculated from the recorded outputs alone. The notebook reports one ANN's metrics, but no baseline predictions, paired errors, confidence intervals, or p-values. The research hypothesis therefore remains untested. Recommended follow-up: train a simple mean predictor and at least one strong tabular baseline (e.g., Random Forest or gradient boosting) using the same train/test split; calculate per-record absolute errors; bootstrap the paired MAE difference over test records; report a 95% confidence interval and p-value if a formal test is prespecified. For deployment, use a vehicle- or trip-grouped test set to avoid leakage and evaluate underestimation errors separately.

```
HYPOTHESIS TESTING RESULTS
--------------------------------------------
Baseline MAE: 2577.97 km
Neural Network MAE: 1176.36 km
Mean MAE improvement: 1401.61 km
Wilcoxon test statistic: 27593.0000
P-value: 0.000000
Significance level: 0.05

Decision: Reject the null hypothesis (H0).
The neural network has statistically lower absolute errors
than the mean-RUL baseline on this test set.
```

Major Findings:

1. The supplied simulated dataset contains 3,000 records and seven selected input features for continuous RUL regression.
2. The ANN achieved MAE = 1,175.77 km, RMSE = 1,457.99 km, and R^2 = 0.7912 on the saved 600-record test set.
3. The example prediction was 6,339.58 km against an actual value of 7,014.84 km, an underestimation of about 675.26 km.
4. Absolute error ranged from 3.23 km to 4,418.35 km, indicating meaningful variability in prediction quality.
5. The notebook saves the model, fitted scaler, and feature list for later reuse.
6. The experiment does not establish real-world accuracy, superiority over other algorithms, or statistical significance.

Recommendations for Industry and System Improvement

1. Do not use the current prototype as the sole basis for brake replacement or safety decisions; retain manufacturer service intervals and physical inspection.
2. Collect real-world measurements across multiple vehicles, riders, loads, road types, weather, and brake component types; include measured pad thickness or a clearly defined end-of-life label.
3. Split data by vehicle, trip, or time rather than relying only on a random row split, and reserve an external test fleet for final evaluation.
4. Benchmark ANN against linear regression, Random Forest, gradient boosting (e.g., XGBoost/LightGBM where available), and temporal models only if sequential sensor histories exist.
5. Report underestimation separately from overestimation because overestimating remaining brake life can be safety-critical.
6. Add uncertainty intervals and conservative alert thresholds; send low-confidence or near-limit cases for inspection.
7. Monitor model drift after deployment and retrain only through a documented validation process.

CONCLUSION

This study implemented a feed-forward ANN to predict two-wheeler brake remaining useful life from seven telemetry and usage features. The notebook's saved evaluation reports MAE of 1,175.77 km, RMSE of 1,457.99 km, and R^2 of 0.7912. The figures and error summaries provide an initial view of model fit and prediction variability. The objectives of building the prototype and evaluating its held-out regression performance were addressed. However, the research questions concerning superiority over baselines and statistical significance were not resolved, because no comparator model or inferential test is present. Most importantly, the dataset is identified as simulated. Real-world deployment requires representative measured data, group-wise external validation, uncertainty estimation, and safety-focused evaluation.

LIMITATIONS OF RESEARCH:

1. The dataset is simulated, and its relationship between inputs and target may not reflect real brake degradation.
2. The random row split may overestimate generalization if records share vehicle, trip, or simulated-generation patterns.
3. The target label's construction and physical ground-truth measurement process are not documented in the notebook.

Future Scope of Research:

1. Build a real-world brake wear dataset with repeated measurements and component replacement records.
2. Use grouped and time-ordered validation to evaluate generalization to unseen
3. Compare tree ensembles and regularized regression with ANN; test CNN/LSTM/Transformer models only when meaningful time-series windows are available.
4. Investigate physics-informed features such as estimated friction-brake work, vehicle mass, braking events, and thermal exposure.
5. Estimate prediction uncertainty with bootstrap ensembles or quantile regression and construct calibrated prediction intervals.
6. Conduct prospective fleet trials and evaluate maintenance outcomes, missed warnings, false alarms, and cost-benefit trade-offs.

REFERENCES:

[1] 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.

[2] H. Li, Z. Zhang, T. Li, and X. Si, "A review on physics-informed data-driven remaining useful life prediction: Challenges and opportunities," Mechanical Systems and Signal Processing, vol. 209, art. 111120, 2024, doi: 10.1016/j.ymssp.2024.111120.

[3] X. Qiao, V. L. Jauw, L. C. Seong, and T. Banda, "Advances and limitations in machine learning approaches applied to remaining useful life predictions: a critical review," The International Journal of Advanced Manufacturing Technology, vol. 133, pp. 4059–4076, 2024, doi: 10.1007/s00170-024-14000-0.

[4] S. Iqbal, M. Bhambri, and R. Lahase, "Brake Pad Life Monitoring System Using Machine Learning," SAE Technical Paper 2024-26-0032, 2024, doi: 10.4271/2024-26-0032.

[5] "Prediction of brake pad wear and remaining useful life considering varying vehicle mass and an experimental holistic approach," Wear, vols. 552–553, art. 205433, 2024, doi: 10.1016/j.wear.2024.205433.