Aircraft Engine Predictive Maintenance

Graduation Project Report

Supervisor: [Eng. Heba Mohamed]
Date: April 2025

Abstract

This project presents the development of a Predictive Maintenance system for industrial equipment using machine learning techniques. The primary goal is to predict the Remaining Useful Life (RUL) of machines, allowing early detection of potential failures and reducing unexpected downtime.

In this work, the **NASA Turbofan Jet Engine Degradation Simulation Dataset** was used, which contains time-series sensor data from **100 aircraft engines** operating under different environmental and operational conditions. The dataset was carefully preprocessed using techniques such as feature selection, scaling, moving averages, and rate of change calculations to prepare it for model training.

Multiple machine learning models were explored, including **Random Forest**, **XGBoost**, and **LightGBM**. Hyperparameter tuning was performed using **Randomized SearchCV** to select the best-performing model. The chosen model demonstrated strong predictive power, effectively estimating the RUL based on the provided sensor measurements.

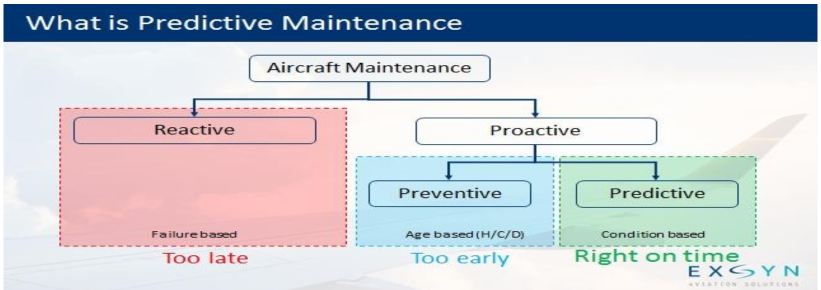
Finally, the system was deployed as an interactive and user-friendly web application using **Streamlit**, allowing real-time predictions and visualization of Remaining Useful Life for new engine data. This project offers a practical solution for industries seeking to implement Predictive Maintenance strategies, ultimately enhancing safety, reliability, and operational efficiency.

1.Introduction

1.1. The Journey of Maintenance Development

In the past, maintenance was typically performed only after equipment had completely failed, which often reduced short-term costs but led to serious problems such as unexpected breakdowns, production delays, and safety risks. To address these issues, industries adopted **Preventive Maintenance**, where machines are serviced at fixed intervals to avoid failure. However, this method doesn't consider the actual condition of the equipment, often resulting in unnecessary maintenance or missed failures.

To solve these challenges, **Predictive Maintenance (PdM)** was introduced, using real-time data and machine learning to predict failures before they happen. This enables timely maintenance, reduces costs, prevents unexpected downtime, and extends equipment life.



1.2.Idea

Our project is centered around implementing **Predictive Maintenance for Aircraft Engines**. We utilize **NASA's Turbofan Jet Engine Data Set (FD001)**, which simulates 21 sensors readings collected from 100 aircraft engines under one operating condition (Sea Level) and one fault mode (HPC Degradation). The goal is to predict the **Remaining Useful Life (RUL)** of each engine — an essential metric that tells how many cycles remain before the engine reaches failure.

The RUL is calculated by subtracting the current cycle count of the engine from its maximum cycle count observed in the dataset. Using this target, we trained machine learning models to predict the future RUL based on historical sensor data, allowing maintenance actions to be taken before failure occurs.

To support real-world usability, we also integrated the model into a user-friendly web interface built with **Streamlit**, enabling engineers to input real-time sensor data and receive RUL predictions instantly.

This predictive approach aims to enhance **aircraft safety, reduce maintenance costs, optimize resource planning**, and minimize unexpected failures, making it a vital component of modern aviation maintenance systems.

1.3. Objectives

The main objectives of this project are:

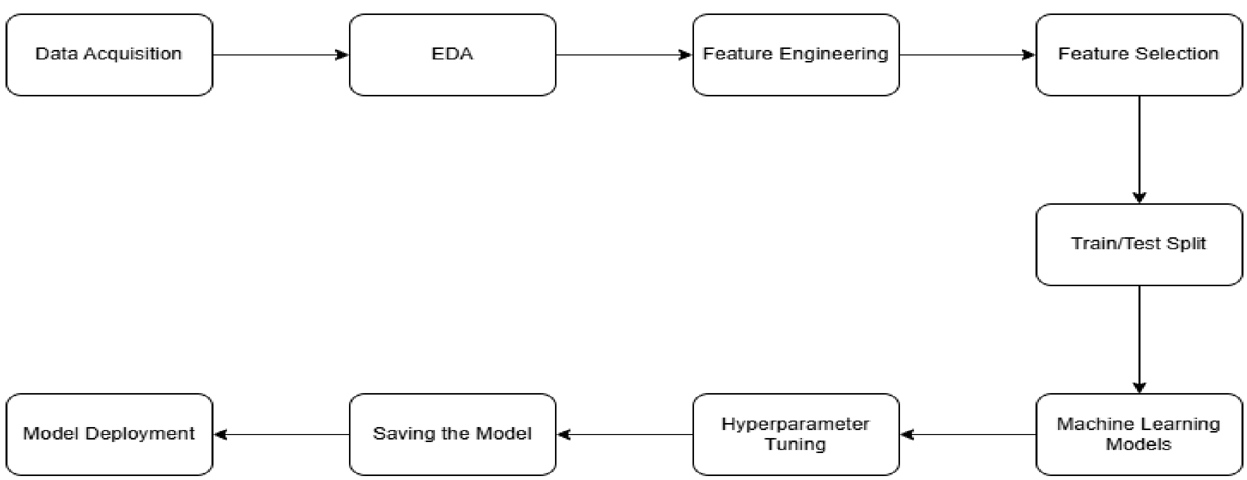
- **Develop a data-driven predictive model** that can accurately forecast the Remaining Useful Life (RUL) of an aircraft engine using multivariate time-series sensor data.
- **Apply feature engineering techniques** including moving averages and rate of change to extract meaningful trends from raw sensor readings.

- **Optimize and compare multiple machine learning algorithms** (such as Random Forest, XGBoost, and LightGBM) to select the best-performing model for RUL prediction.
- **Deploy the final model using Streamlit** to create an interactive and user-friendly web application that allows users to input new sensor data and instantly predict the engine's remaining life.
- **Validate and test the model** using the NASA **FD001** subset, which contains:
  - 100 engines in training data.
  - 100 engines in test data.
  - 21 sensors
  - Operating under one condition (**Sea Level**).
  - Degrading under one fault mode (**HPC Degradation**).

By achieving these objectives, the project aims to offer real-time insights that help in **avoiding unexpected engine failures**, reducing unnecessary maintenance activities, and ultimately **enhancing safety and operational reliability**.

3.Methodoligy

This section outlines the methodology used to develop the predictive maintenance (PdM) model, including data Acquisition , preprocessing, feature engineering, feature Selection, split the data, ML model selection, Hyperparameter , training, and evaluation. The goal is to predict equipment failures or estimate remaining useful life (RUL) using sensor data and machine learning techniques.



Model Architecture

3.1.Data Acquisition

We used the **NASA Turbofan Jet Engine Degradation Simulation Data Set (FD001)**, which contains multiple engine run-to-failure records. The data includes sensor readings for 100 engines in training and another 100 in testing, under one operating condition (Sea Level) and one fault mode (High-Pressure Compressor degradation).

	unit_nr	time_cycles	setting_1	setting_2	setting_3	s_1	s_2	s_3	s_4	s_5	s_6	s_7	s_8	s_9	s_10	s_11	s_12	
0	1	1	-0.0007	-0.0004	100.0	518.67	641.82	1589.70	1400.60	14.62	21.61	554.36	2388.06	9046.19	1.3	47.47	521.66	238
1	1	2	0.0019	-0.0003	100.0	518.67	642.15	1591.82	1403.14	14.62	21.61	553.75	2388.04	9044.07	1.3	47.49	522.28	238
2	1	3	-0.0043	0.0003	100.0	518.67	642.35	1587.99	1404.20	14.62	21.61	554.26	2388.08	9052.94	1.3	47.27	522.42	238
3	1	4	0.0007	0.0000	100.0	518.67	642.35	1582.79	1401.87	14.62	21.61	554.45	2388.11	9049.48	1.3	47.13	522.86	238
4	1	5	-0.0019	-0.0002	100.0	518.67	642.37	1582.85	1406.22	14.62	21.61	554.00	2388.06	9055.15	1.3	47.28	522.19	238

3.2.preprocessing

- **Missing Values & Duplicates:** Checked and none found.

- **Feature Engineering** :

  - Computed **RUL** per unit by subtracting current time cycle from the engine's max cycle.
  - **Rate of Change (ROC):**Calculated the difference between the current sensor value and the value from the previous cycle to highlight rapid shifts in sensor readings.
  - **1 - Absolute Rate of Change**: Computed the absolute difference between a sensor's current value and its previous value, helping the model detect sharp changes regardless of direction.
  - **2 - Relative Change (Percentage Change) :**Calculated the percentage difference from the previous cycle to normalize changes across varying scales and emphasize relative deviations in sensor behavior.
  - **3 - Rolling Change over a Window**: Determined the trend of each sensor across a sliding window of cycles to capture the long-term degradation pattern of the engine.
  - **Moving Average**: Applied a moving average filter on each sensor to smooth out short-term fluctuations and highlight long-term trends in the data.

- Scaled numerical features using **MinMaxScaler.**

- Clipping of RUL at 125 as after 125 , RUL is responding to the sensor values after this value. This is done to improve performance of the applied models .

3.3.ERD

5. Exploratory Data Analysis (EDA)

Key Visuals & Insights:

- **RUL Trend Analysis** for Top Units
- **Sensor Behavior Over Time**
- **Distribution of Engine Lifespan**
- Clear sensor degradation patterns were observed over cycles.

3.4.Feature Selection

Correlation Analysis:
A correlation heatmap was created to detect and eliminate features that showed a weak relationship with RUL or were highly correlated with each other (multicollinearity), which could mislead the model.

During the analysis phase, we used **heatmaps** and **plots** to explore sensor behavior. It was found that the following sensors: **{ s1, s5, s6, s10, s16, s18, s19 }**

showed no significant variation across cycles. Since constant features do not contribute useful information for prediction, they were excluded from the model to improve efficiency and accuracy.

Additionally, **Principal Component Analysis (PCA)** was applied to reduce feature dimensionality and remove redundant information. PCA helped identify the most influential components affecting RUL prediction and supported the selection of different sensor combinations during model training.

3.5. Model Selection

In order to predict the **Remaining Useful Life (RUL)** of aircraft engines, we implemented and evaluated several regression models. The models used include:

- **Linear Regression**
- **K-Nearest Neighbors Regressor**
- **Decision Tree Regressor**
- **Random Forest Regressor**
- **XGBoost Regressor**
- **LightGBM Regressor**

Each model was trained and evaluated to select the one that provides the best performance for accurate RUL prediction.

Pipeline:

- Scikit-learn Pipeline used for modular model building.
- Extensive **hyperparameter tuning** with GridSearchCV.

3.6. Evaluation Metrics

- **Mean Absolute Error (MAE)**
- **Mean Squared Error (MSE)**
- **R2 Score**
- Visualization of predicted vs actual RUL values

Best model selected based on **R^2 score and generalization**.

3.7. Deployment & Visualization

The final trained model was deployed using **Streamlit**, allowing interactive visualization and real-time predictions for new engines. This makes the predictive maintenance system accessible and user-friendly for maintenance teams.

4.Result

In order to evaluate the performance of different machine learning models for predicting the Remaining Useful Life (RUL) of aircraft engines, we trained and tested several regression models on the NASA Turbofan Jet Engine Dataset. Initially, the models were trained **without applying any clipping** to the RUL values in the y_train set.

The following table shows the performance results in terms of **R^2 Score**:

Regression Models before clipping

	model name	r2_score
0	Linear Regression	0.357816
5	LightGBM	0.315035
3	Random Forest Regressor	0.269925
1	knn regressor	0.212828
4	XGB Regressor	0.166207
2	Decision Tree	-0.580907

```
xgb_best_model = xgb_random_search.best_estimator_
xgb_y_pred = xgb_best_model.predict(X_test)
print("XGBoost RMSE on Test Set:", np.sqrt(mean_squared_error(y_test, xgb_y_pred)))
print("XGBoost r2 Score on Test Set:", r2_score(y_test, xgb_y_pred))
```

```
XGBoost RMSE on Test Set: 34.2122889162743
XGBoost r2 Score on Test Set: 0.32219505310058594
```

xgboost before clipping with hyperparameter:

However, after analyzing the dataset, we noticed that extreme values for RUL might negatively affect model training, especially for engines that had **RUL values higher than 125**, which in real-world scenarios are not practically relevant. So we applied the following clipping technique:

y_train = y_train.clip(upper=125)

This step limited the RUL values to a maximum of 125 and improved the model stability and prediction accuracy.

After clipping, the models were retrained and the results significantly improved:

	model name	r2_score
3	Random Forest Regressor	0.770434
4	XGB Regressor	0.758420
0	Linear Regression	0.695945
1	knn regressor	0.676723
2	Decision Tree	0.491925

Regression Models after clipping

Based on the improved performance, especially for **Random Forest Regressor, LightGBM, and XGB Regressor**, we selected these three models for further optimization using **RandomizedSearchCV** to fine-tune their hyperparameters.

This step ensured that we achieve the best possible predictive performance for estimating the Remaining Useful Life (RUL) in real-world scenarios.

```
LightGBM RMSE on Test Set: 20.197746606555825
LightGBM r2 Score on Test Set: 0.7637638835134758
```

LightGBM hyperparameter tuined

```
RandomForestRegressor RMSE on Validation Set: 20.107050168795755
RandomForestRegressor r2 Score on Test Set: 0.7658807205301029
```

Random forest hyperparameter tuined

```
XGBoost RMSE on Test Set: 19.75053385725268
XGBoost r2 Score on Test Set: 0.7741093635559082
```

XGBoost hyperparameter tuined

Based on the evaluation results after applying RUL clipping, we observed that the **XGBoost Regressor** achieved the highest **R^2 Score** among all the tested models, reaching a performance of **0.774109**.

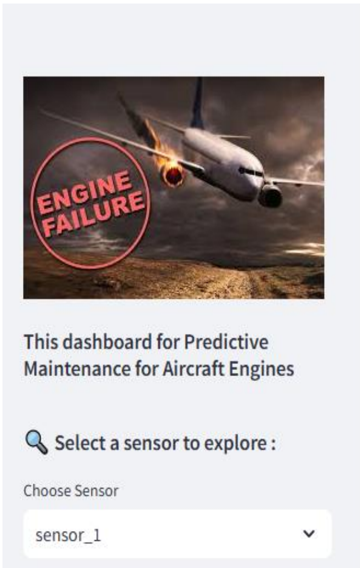
Therefore, we selected the **XGBoost Regressor** as our final model for predicting the Remaining Useful Life (RUL) of aircraft engines, as it provided the best balance between accuracy, robustness, and generalization on unseen data.

5.Model Deployment

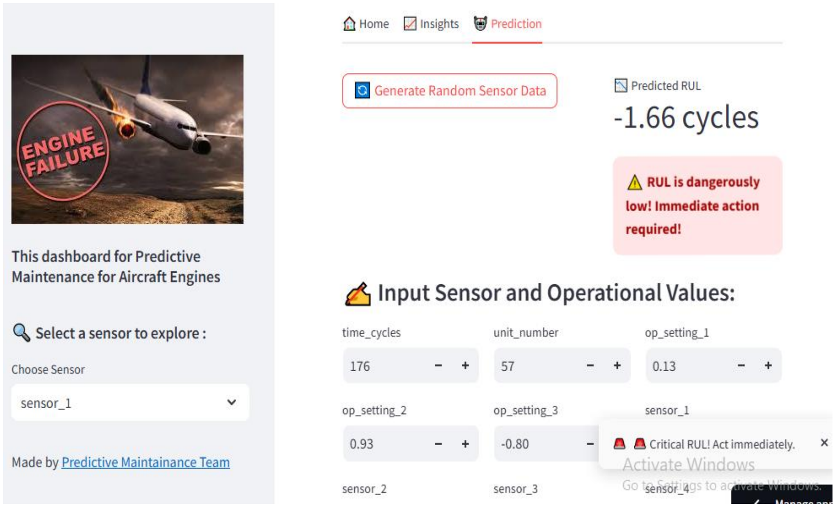
The final trained model was deployed using a **Streamlit** dashboard enable real-time interaction and visualization. The dashboard was designed to be user-friendly and intuitive, helping maintenance teams easily access predictions and insights.

Tabs:

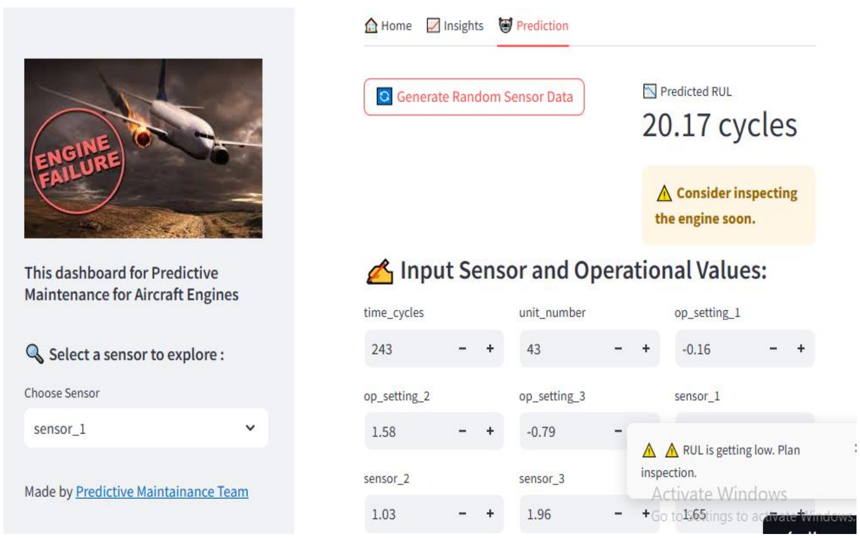
- **Home:** Introduction, visual story using aircraft images

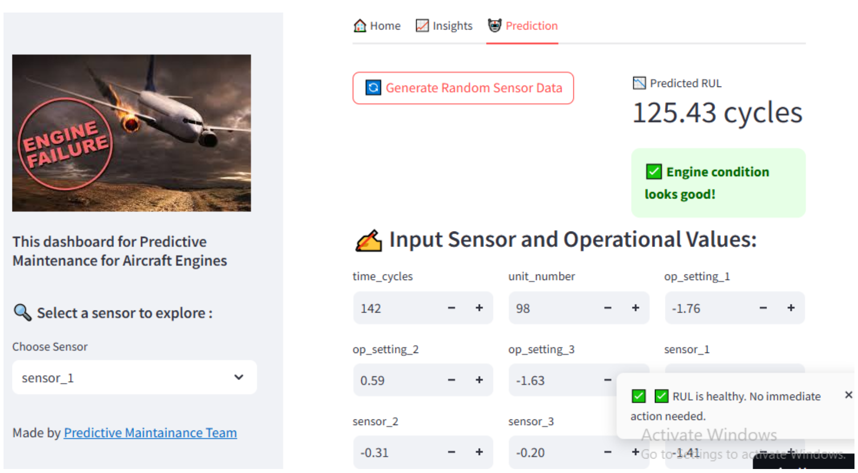


- **Insights:** Interactive trend plots of selected sensors using Plotly
- **Prediction:**
  - Manual or random input generation
  - Real-time RUL prediction
  - Critical alerts with animated visual feedback



Predicted RUL Feedback & Alerting System





Predicted RUL Feedback & Alerting System Figures

6.Future Work

- Explore deep learning models (LSTM, GRU)
- Integrate with real-time IoT sensor data
- Add user authentication and model update API
- Deploy on cloud with scheduled retraining pipeline

7.Conclusion

This project successfully demonstrated the power of Predictive Maintenance in forecasting the Remaining Useful Life (RUL) of aircraft engines using real-time sensor data and machine learning models. By leveraging the NASA Turbofan Jet Engine Dataset (FD001), we were able to simulate real-world engine degradation and develop a robust predictive model.

Throughout the project, different machine learning algorithms were evaluated, and through careful preprocessing, feature engineering, and hyperparameter tuning, the **XGBoost Regressor** was selected as the final model due to its superior prediction performance.

The predictive maintenance system was deployed using **Streamlit**, providing a user-friendly and interactive web application for real-time RUL predictions and visualization of sensor trends. This system has the potential to significantly reduce unplanned downtime, optimize maintenance scheduling, and enhance both safety and operational efficiency.

Overall, this project highlights the importance of combining data science techniques with domain knowledge to solve real industrial problems and opens the door for future development, including the integration of deep learning models and real-time IoT data.

Appendix

- GitHub Repo: [https://github.com/KarimXHamed/PredictiveMaintenance]
- Streamlit :[Streamlit]
- Dataset: NASA CMAPSS FD001
- Tools: Python, Scikit-learn, XGBoost, MLflow, Streamlit, Plotly

Team Members:

Fatma Al-Zahraa Hassan Abdullah

Karim Hamed Ashour

Omayma Ali Abdelhafeez

Rana Ahmed Abu Gharib

Reham Ashraf Mohammed