Predictive Maintenance Project Using NASA Turbofan Engine Dataset

By: Mamadou Ndiong

Introduction

Industrial machinery failures cause costly downtime, emergency repairs, and safety hazards.

The goal was to build a Predictive Maintenance System that can predict Remaining Useful Life (RUL) of industrial engines, allowing maintenance to be scheduled before failures occur.

I used the NASA Turbofan Jet Engine Degradation Simulation Dataset for this project.

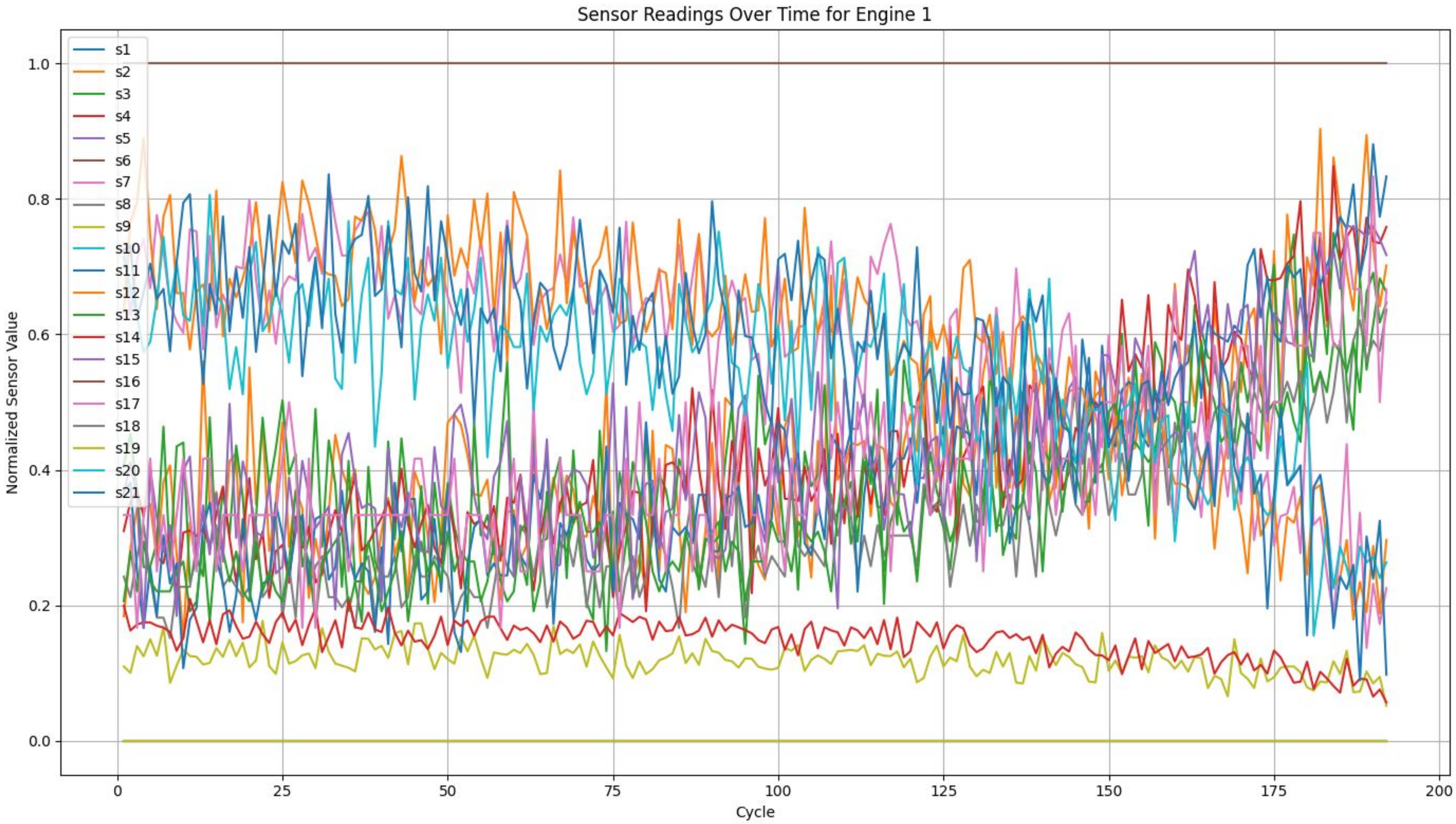
Dataset Overview

I worked with three key files:

- train_FD001.txt: Full operational sensor data until failure (for training).
- test_FD001.txt: Sensor data up to a certain time (engines haven't failed yet).
- RUL_FD001.txt: True RUL values for the test engines.

Each record included:

- 3 operational settings,
- 21 sensor readings,
- An engine ID and its cycle number.


Sensor Readings Over Time for Engine 1
s1
s2
s3
s4
s5
s6
s7
s8
s9
s10
s11
s12
s13
s14
s15
s16
s17
s18
s19
s20
s21
Normalized Sensor Value
Cycle
0.0
0.2
0.4
0.6
0.8
1.0
0
25
50
75
100
125
150
175
200

Phase 1: Data Preprocessing

Steps:

- Assigned column names to the dataset for clarity (e.g., setting1, s1, ..., s21).
- Checked and confirmed no missing values.
- Analyzed sensor variance:
  - Removed sensors that showed little to no variance and thus carried little useful information.
- Scaled features:
  - Used MinMaxScaler to normalize sensor readings between 0 and 1 (essential for ML models).

Feature Engineering

Key additions:

- Normalized Cycle Life (cycle_norm):
  - Represented how far into its life an engine was (0 = new, 1 = near failure).
- Rolling Features:
  - Created 5-cycle rolling mean and rolling standard deviation for each key sensor to capture local trends.
- Delta Features:
  - Captured the change between current and previous cycle sensor readings (important for spotting sudden deterioration).

Feature engineering expanded the dataset with trend-sensitive and change-sensitive inputs to help the model better anticipate failures.

Phase 3: Modeling - RUL Prediction

Model Trained:

- Random Forest Regressor

Training Setup:

- Input Features: All engineered features (excluding ID and cycle).
- Target: Remaining Useful Life (RUL) at each cycle.

Performance on Training Set:

- MAE ≈ 5.47 cycles
- RMSE ≈ 8.11 cycles

The model fit the training data extremely well.

Phase 4 - Model Evaluation on Test Set

1. For each test engine, we took only the last recorded cycle.
2. Predicted its RUL using the trained model.
3. Compared predictions to the true RUL from RUL_FD001.txt.

Test Set Results:

- MAE ≈ 75.5 cycles
- RMSE ≈ 86.2 cycles

Observations and Challenges

The model overfit to training data (dense, complete histories) but struggled when only predicting based on a single snapshot (test set).

Predicting RUL is inherently harder at early life stages when failures are not yet obvious in the data.

Going Forward

Improve Random Forest:

- Use smarter sampling (train only on late-life data).
- Apply cross-validation at the engine-level.

Try Sequential Models:

- Explore LSTM or GRU models to capture the temporal dynamics more accurately.

Feature Optimization:

- Further refine feature selection to remove noise and redundancy.