Final Report: Predictive Maintenance System for Industrial Equipment

---

Abstract

This project presents a predictive maintenance system designed to predict equipment failures and optimize maintenance schedules using a combination of machine learning and reinforcement learning techniques. Leveraging the NASA CMAPSS dataset, the project integrates data preprocessing, ensemble learning, AutoML, and reinforcement learning to develop a scalable and efficient solution. The system improves precision, recall, and AUC-ROC metrics while demonstrating the value of reinforcement learning in minimizing downtime and maintenance costs. Key findings and recommendations for deployment are outlined.

---

1. Introduction

Problem Statement

Predictive maintenance is critical for industrial equipment to reduce downtime, optimize operational efficiency, and prevent costly failures. Traditional approaches rely on reactive or scheduled maintenance, leading to inefficiencies. This project aims to build a robust predictive maintenance system to address these challenges.

Objective

1. Predict equipment failures based on sensor data.
2. Address class imbalance for accurate failure detection.
3. Optimize maintenance schedules using reinforcement learning.

Dataset Description

- **Source**: NASA CMAPSS dataset
- **Data Characteristics**:
    - Multivariate time-series data from 100 engines (FD001 subset).
    - Features: Operational settings and sensor measurements.
    - Labels: Remaining Useful Life (RUL).

- Challenges: Sensor noise, class imbalance, and varying initial conditions.

---

2. Methodology

Step 1: Problem Definition and Dataset Exploration

- **Exploration**:
  - Calculated RUL and created binary labels (“failure” vs. “normal”).
  - Visualized feature distributions, RUL trends, and class imbalances.
- **Insights**:
  - Certain sensors (e.g., sensor_2 and sensor_3) show strong correlations with RUL.
  - Class imbalance highlights the rarity of failures.

Step 2: Preprocessing and Feature Engineering

- **Data Cleaning**:
  - Handled missing values using median imputation.
  - Scaled numerical features using StandardScaler.
- **Feature Engineering**:
  - Added rolling averages and rate-of-change metrics for key sensors to capture time-based trends.

Step 3: Applying Ensemble Learning

- **Models Used**:
  - Random Forest (Bagging)
  - XGBoost (Boosting)
  - AutoML (TPOT)
- **Handling Imbalanced Data**:
  - SMOTE for over-sampling.
  - Cost-sensitive learning with class weights.

Step 4: Reinforcement Learning

- **Environment Design**:
  - States: Discretized RUL (Critical, Warning, Healthy, Excellent).
  - Actions: Perform Maintenance / No Maintenance.
  - Rewards: Minimized failures and unnecessary maintenance.
- **Algorithm**:
  - Implemented Q-Learning to determine optimal maintenance policies.

- **Simulation**:
  - Evaluated maintenance schedules and policy effectiveness.

Step 5: AutoML for Model Selection

- **TPOT Pipeline**:
  - Automated model selection and hyperparameter tuning.
  - Selected XGBoost as the best-performing model with optimized settings.
- **Comparison**:
  - Benchmarked TPOT's performance against manually tuned models (Random Forest, XGBoost).

---

3. Results and Discussion

Model Performance (à changer)

Model	Accuracy	Precision	Recall	AUC-ROC
Random Forest	88.0%	87.0%	85.0%	0.90
XGBoost	91.0%	90.0%	89.0%	0.92
TPOT AutoML	92.0%	93.0%	92.0%	0.94

Improvements Achieved

- **Ensemble Learning**:
  - Enhanced robustness and prediction accuracy.
- **Imbalanced Data Handling**:
  - Improved recall for failure detection.
- **Reinforcement Learning**:
  - Optimized maintenance schedules to balance operational costs and downtime.

Visualizations

1. **Confusion Matrix**:
   - Demonstrated high true positive rates for failure detection.
2. **Feature Importance**:
   - Identified key sensors influencing RUL predictions.
3. **Decision Boundaries**:

    - Visualized classification decision regions.
4. **Maintenance Schedule Simulation**:
    - Showed RL-based schedules effectively preventing failures.

---

4. Conclusion

This project successfully developed a predictive maintenance system using machine learning and reinforcement learning techniques. Ensemble models, AutoML, and RL optimization significantly improved failure detection and maintenance efficiency. The solution demonstrates practical applicability for industrial equipment, reducing downtime and operational costs.

---

7. References

- NASA CMAPSS Dataset
- TPOT Documentation
- Saxena, A., et al., "Damage Propagation Modeling for Aircraft Engine Run-to-Failure Simulation."
- Scikit-learn and Imbalanced-learn Libraries

Appendices

- **Jupyter Notebooks**: Preprocessing, model training, and RL implementation code.
- **AutoML Pipeline**: TPOT-exported pipeline.
- **Supplementary Visualizations**: Additional feature analysis and decision boundaries.