Final Project Report

Title: *PumpSureAI: Predictive Maintenance with Sensor Data*

Introduction

This report summarizes the analysis and modeling work conducted to address the predictive maintenance problem for industrial pumps. The dataset, sourced from Kaggle, contains time-series sensor data recorded from industrial pumps. The primary objective of this project is to identify pump failures before they occur, enabling minimized downtime and optimized maintenance schedules.

Data Summary

The dataset comprises approximately 220,000 rows with the following key features:

- **Time**: Timestamp for each observation.
- **Sensor Readings**: Sensor data(52 series): All values are raw.
- **Pump Status**: Binary indicator of failure or normal operation.
- **Features Summary**:
    - **Numerical Variables**: Sensor readings (e.g., temperature, pressure, vibration).
    - **Target Variable**: Pump status (failure vs. normal).

Data Cleaning and Exploration

- **Handling Missing Data**: Imputed missing sensor readings using time-series techniques and statistical methods (e.g., forward fill, mean imputation).
- **Outlier Treatment**: Applied quantile-based thresholds to remove extreme sensor readings.
- **EDA Insights**:
    - Identified strong correlations between vibration levels and pump failures.
    - Temporal patterns suggest failure likelihood increases under high-pressure fluctuations.

Methodology

Data Preprocessing

- One-hot encoding of categorical variables (e.g., pump type).
- Normalization is applied to numerical features to ensure uniform scaling.

- Train-test split at:

  train_X = df_x[0:130000]   train_Y = df_y[0:130000]

  test_X = df_x[130000::]   test_Y = df_y[130000::]

Modeling Approach

- Implemented the following models:
  1. LSTM-based classification model
  2. Logistic Regression
  3. KNN
  4. SVC
  5. CART
  6. Random Forest
  7. AdaBoost
  8. GBM
  9. XGBoost
  10. LightGBM
- Hyperparameter tuning uses RandomSearch for LSTM and GridSearchCV for other models to optimize performance.

Evaluation Metrics

- Metrics used: Precision, Recall, F1-Score, and Accuracy.

Results

Model Performance Summary

Classification Report:

	precision	recall	f1-score	support
0	1.00	0.99	0.99	32186
1	0.98	1.00	0.99	14484
accuracy			0.99	46670
macro avg	0.99	0.99	0.99	46670
weighted avg	0.99	0.99	0.99	46670

Tuning Results:

LR: Training F1 Score = 0.9904, Test F1 Score = 0.9473

Adaboost: Training F1 Score = 0.9893, Test F1 Score = 0.9914

GBM: Training F1 Score = 0.9979, Test F1 Score = 0.9698

XGBoost: Training F1 Score = 0.9945, Test F1 Score = 0.9806

AdaBoost demonstrated the highest F1-Score, making it the most effective model for this task.

Key Visualizations

1. **Correlation Heatmap**: Highlights relationships among sensor readings, aiding in feature selection.
2. **Sensor Trends Over Time**: Shows patterns of anomalies leading up to failures.
3. **Model Comparison Bar Chart**: Illustrates differences in F1-scores across all models.

Recommendations

1. **Deploy AdaBoost Model**: Utilize the AdaBoost model for real-time monitoring and failure prediction.
2. **Integrate Real-Time Sensor Data**: Enhance predictions with live data streams.
3. **Explore Advanced Techniques**: Investigate deep learning models for complex feature interactions.

Conclusion and Future Work

This project demonstrates the viability of predictive maintenance using machine learning. Future work could include:

- Expanding the dataset with more failure cases.
- Incorporating external data sources, such as environmental conditions.
- Testing advanced algorithms like LSTMs for improved temporal pattern recognition.

Appendix

- Feature engineering methods.
- Model configurations and hyperparameters.
- Source code and full documentation.