Fault Detection of Industrial Gas Turbine Engines

By: Garey Salinas
University of Colorado - Boulder

Introduction

Context: Industrial gas turbines drive compression systems in midstream pipeline operations

Challenge: Unplanned downtime costs millions in lost production

Opportunity: Sensor data enables early fault detection

Approach: Machine learning classification models

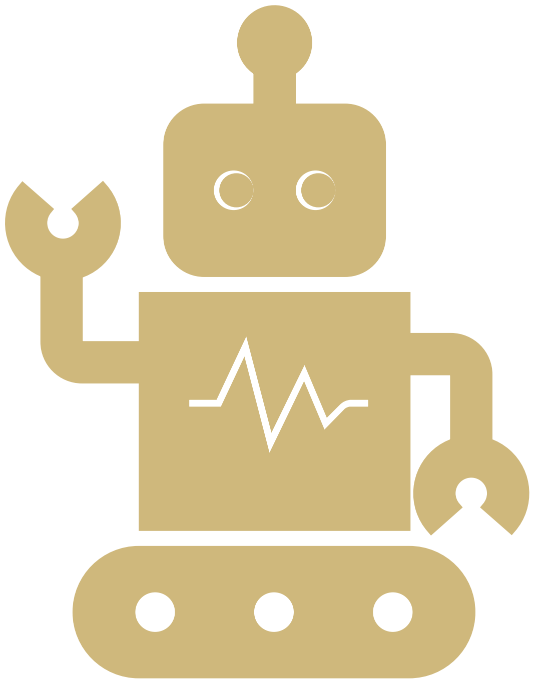
Goal: Develop models that accurately classify gas compressor failures

Executive Summary

- Developed fault classification model using the Kaggle Gas Turbine Engine Fault Detection Dataset
- Built robust preprocessing pipeline including feature scaling and feature engineering
- Applied dimensionality reduction techniques to improve model performance
- Employed multiple classification algorithms with k-fold cross-validation
- Identified key features that contribute most significantly to fault conditions

Problem Statement

- **Business Impact:** Unexpected turbine failures cause significant financial losses
- **Technical Challenge:** Complex sensor data with subtle fault patterns
- **Current Limitations:** Traditional monitoring misses early warning signs
- **Goal:** Develop scalable, interpretable fault detection systems combining supervised classification with unsupervised anomaly detection

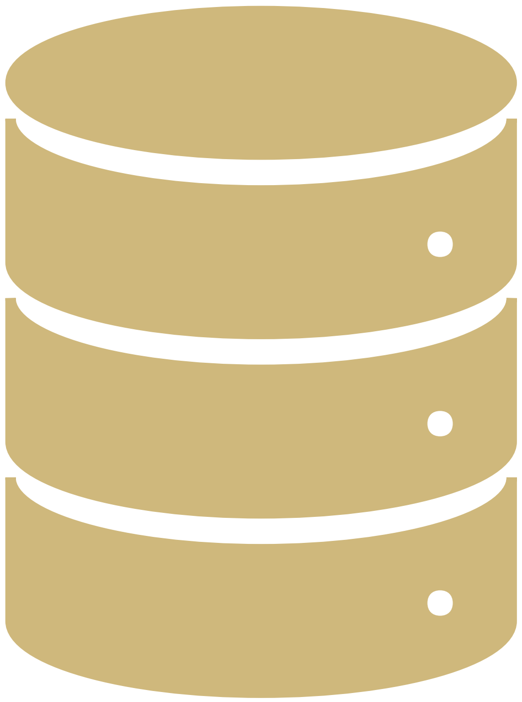


Related Work

- Machine learning techniques gaining traction in industrial maintenance
- IoT Analytics research emphasizes demand for interpretable solutions
- Applied Sciences survey confirms applicability of selected methods
- Deep learning techniques emerging for temporal evolution of faults
- Classical ML models remain valuable for interpretability and robust performance

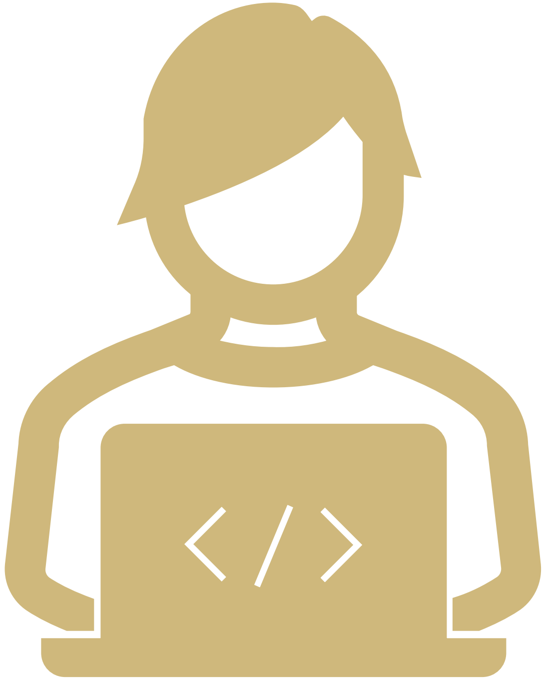
Dataset Overview

- **Source:** Kaggle Gas Turbine Engine Fault Detection Dataset [3]
- **Size:** 1,386 samples with 10 continuous variables
- **Features:** Temperature, torque, pressure, vibration, RPM, power output
- **Target:** Fault indicators (converted to binary classification)
- **Value:** Real-world signals with labeled fault conditions



Methodology Overview

- Data preprocessing and feature scaling
- Feature engineering with domain-specific ratios
- Dimensionality reduction using PCA
- Outlier detection with Isolation Forest and DBSCAN
- Model training and cross-validation



Feature Engineering

- **Standardization:** Applied to prevent features with larger units from dominating
- **Domain-specific ratios:** Created engineered features based on turbine physics
- Pressure ratios
- Power-to-fuel ratios
- **Importance:** Pressure ratios particularly indicative of early fault conditions
- **Benefit:** Provides actionable insights for domain experts

Dimensionality Reduction with PCA

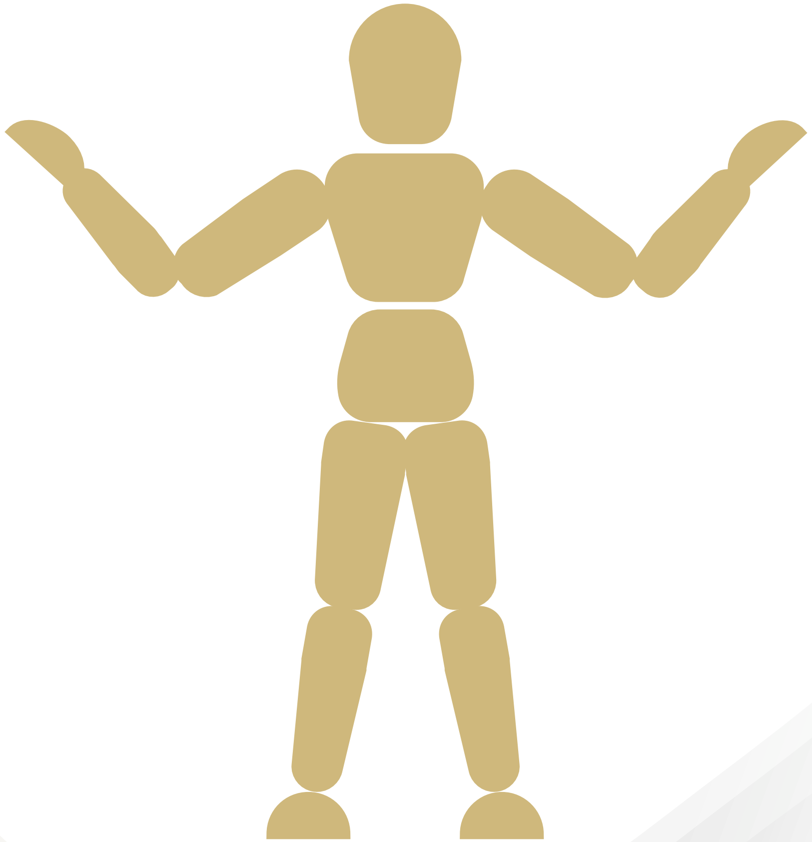
- Reduced feature space dimensionality while preserving information
- Retained principal components that explain 95% of variance
- Improved model training efficiency and generalization
- Enhanced visualization of class separability
- Reduced model complexity for better interpretability

Outlier Detection

- **Methods Used:**
  - Isolation Forest: Identifies anomalies through random feature splitting
  - DBSCAN: Density-based spatial clustering to find operational anomalies
- **Application:** Outliers used to create new features added to dataset
- **Benefit:** Combined results enhance robustness of anomaly detection

Modeling Techniques

- **Logistic Regression:** Baseline model with high interpretability
- **Random Forest Classifier:** Ensemble method robust to overfitting
- **Support Vector Machines (SVM):** Effective for machinery fault detection
- **XGBoost:** Gradient boosting for high performance classification

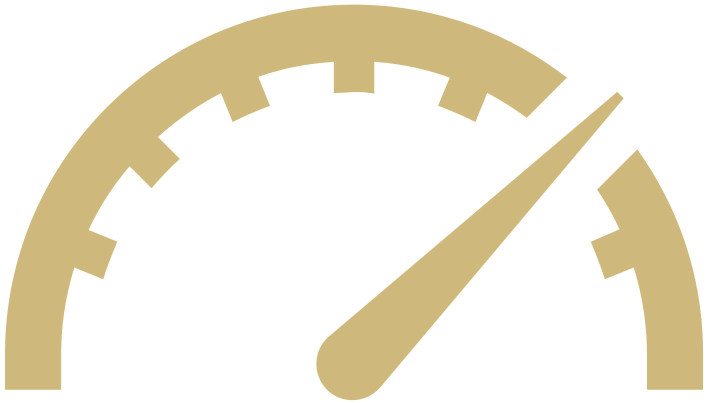


Validation Approach

- K-fold cross-validation for robust performance estimation
- Split data into training and testing sets
- Hyperparameter tuning via grid search
- Feature importance analysis for each model
- **Performance metrics:** accuracy, precision, recall, F1-score

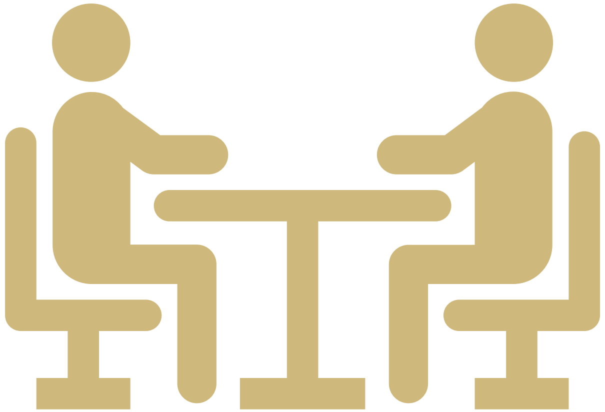
Evaluation Results

- **Model Performance Comparison:**
  - SVM and Random Forest showed highest effectiveness for machinery fault detection
  - Logistic Regression provided most interpretable results
  - Combined approaches leveraged strengths of multiple algorithms
- **Key Features:** Pressure ratios, temperature, and vibration measurements most predictive



Feature Importance Analysis

- **Key Contributing Features:**
  - Temperature measurements
  - Pressure ratios
  - Vibration readings
- Engineered features vs. raw sensor data
- **Insights:** Physical interpretation of the most significant features



Discussion

Project Evolution

- Initially aimed for time-series analysis, pivoted to static data classification due to dataset availability

Key Findings

- Combined outlier detection methods (Isolation Forest + DBSCAN) enhanced anomaly identification
- Random Forest and XGBoost showed highest accuracy
- Engineered features significantly outperformed raw sensor data

Model Insights

- Feature importance analysis revealed critical monitoring parameters
- PCA improved visualization while maintaining 95% variance
- Results provide clear maintenance inspection priorities

Conclusion

- Successfully developed fault detection system for industrial gas turbines
- Preprocessing pipeline with feature engineering, PCA, and outlier detection
- Multiple models evaluated with k-fold cross-validation
- System supports proactive maintenance decision-making
- Potential to minimize unexpected downtime and improve operational efficiency

Timeline and Milestones

18 Apr.
Proposal & literature review

20 Apr.
Data cleaning, Feature Engineering and EDA

21 Apr.
Isolation Forest, DBSCAN

23 Apr.
Train RF, SVM, XGBoost, Logistic Regression

28 Apr.
Final report & presentation

Next Steps

- Further refine feature engineering based on domain expertise
- Explore additional classification algorithms
- Develop visualization tools for real-time monitoring of key features
- Integrate findings with maintenance scheduling systems
- Extend approach to other types of rotating machinery

Thank You

github.com/gareytwin1/gas-turbine-fault-detection/