Fault Detection of Industrial Gas Turbine Engines

Garey Salinas
University of Colorado-Boulder

Introduction

- This notebook analyzes sensor data from industrial gas turbine engines to detect faults and support predictive maintenance.
- The **main goal** is to develop machine learning models that can accurately classify gas compressor failures. This is critical for midstream pipeline operations, where turbines drive compression systems.

Expected Impact and Applications

- Reduced downtime, optimized maintenance, cost savings.
- Enhanced safety in gas turbine operations.
- Applications: Pipelines, power plants, manufacturing.

Literature Review

- SVM and Random Forests are effective for machinery fault detection.
- Pressure ratios, temperature, and vibration are key features.
- Isolation Forest and DBSCAN help detect operational anomalies.
- PCA improves interpretability and reduces model complexity.

Dataset Overview and Sources

- Source: Kaggle Gas Turbine Engine Fault Detection Dataset
- 1,386 samples: sensor readings (temperature, pressure, RPM).
- Supervised: Includes labeled operational/faulty states.
- Dataset: https://www.kaggle.com/datasets/ziya07/gas-turbine-engine-fault-detection-dataset

Data Preprocessing Techniques

- The dataset had no missing values.
- Converted fault labels to boolean.
- Scaled features using StandardScaler.
- Created features like pressure ratios, flow/RPM.
- Applied PCA to reduce noise.
- Removed outliers with Isolation Forest & DBSCAN.

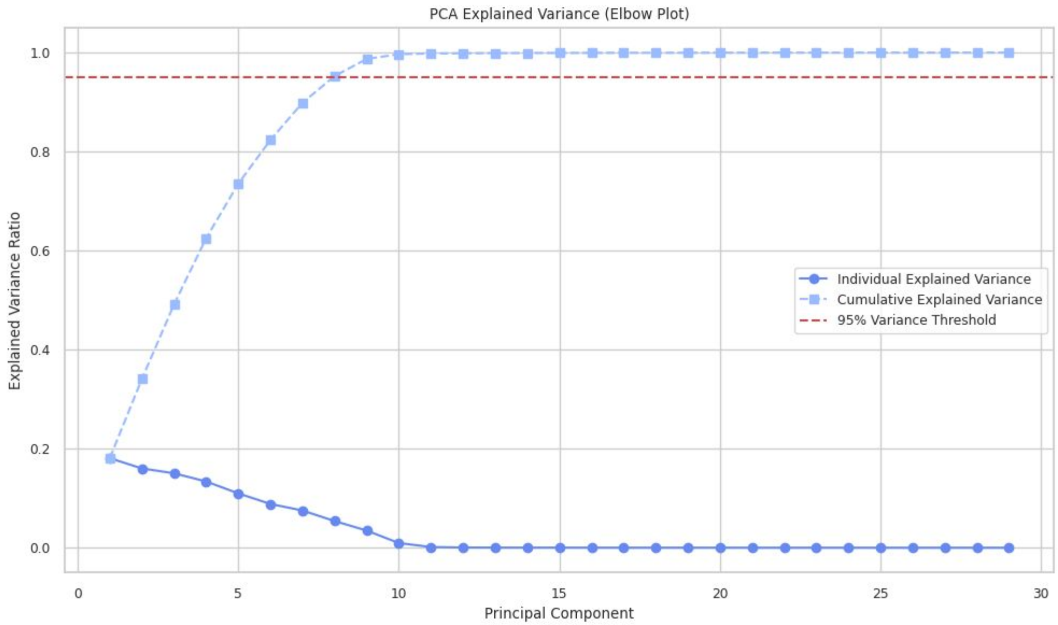
Feature Scaling

- StandardScaler was employed to normalize the sensor readings.
- This ensures that all features contribute equally to the model training.
- It prevents features with larger units (e.g., RPM) from dominating the results.
- Scaling is applied before conducting PCA and fitting the model.

Dimensionality Reduction with PCA

- PCA was applied to reduce feature space dimensionality.
- Retained principal components that explain most variance.
- Helps visualize data clusters and class separability.
- Improves model training efficiency and generalization.
- PCA applied after scaling to ensure correct component weighting

PCA Explained Variance



- **Steep Climb (PC1 to PC6)**
  - First ~6 components explain most of the variance.
  - Each contributes significant new information.
- **Elbow Point (~Component 6 or 7)**
  - After PC6, additional components add little value.
  - "Elbow" indicates diminishing returns.
- **95% Threshold Reached at PC8**
  - The cumulative curve crosses 95% at PC8.
  - First 8 components retain ≥ 95% of total variance.

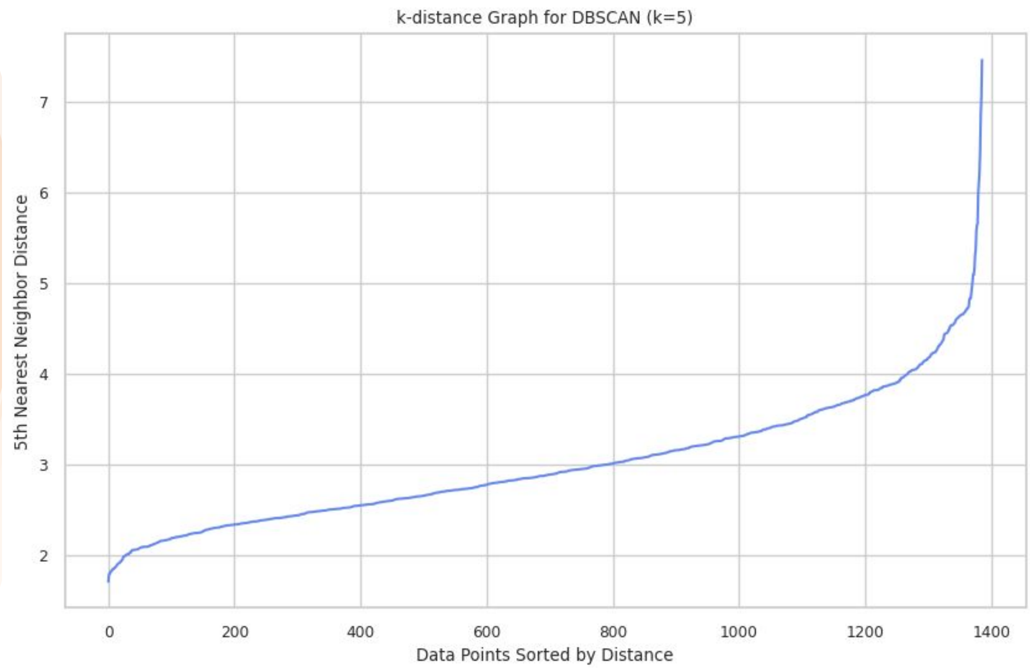
Outlier Detection

- **Goal**: Identify sensor readings that deviate significantly from normal operating behavior.
- **Methods Used**:
  - **Isolation Forest**: Detects anomalies by randomly isolating observations in tree structures.
  - **DBSCAN**: Density-based clustering that flags low-density points as outliers.
- **Why It Matters**:
  - Removes noisy or anomalous data before modeling.
  - Enhances model accuracy and reduces false positives.

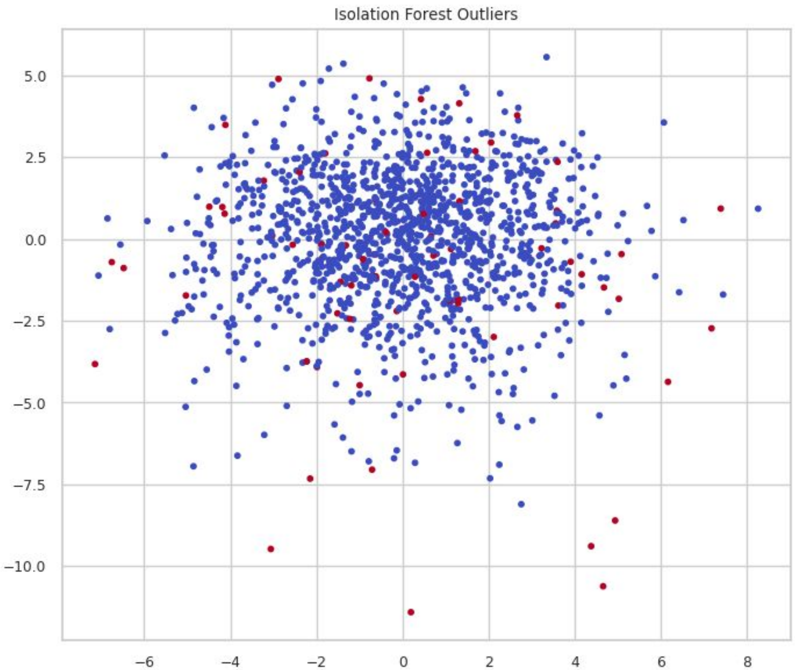
Outlier Detection
DBSCAN

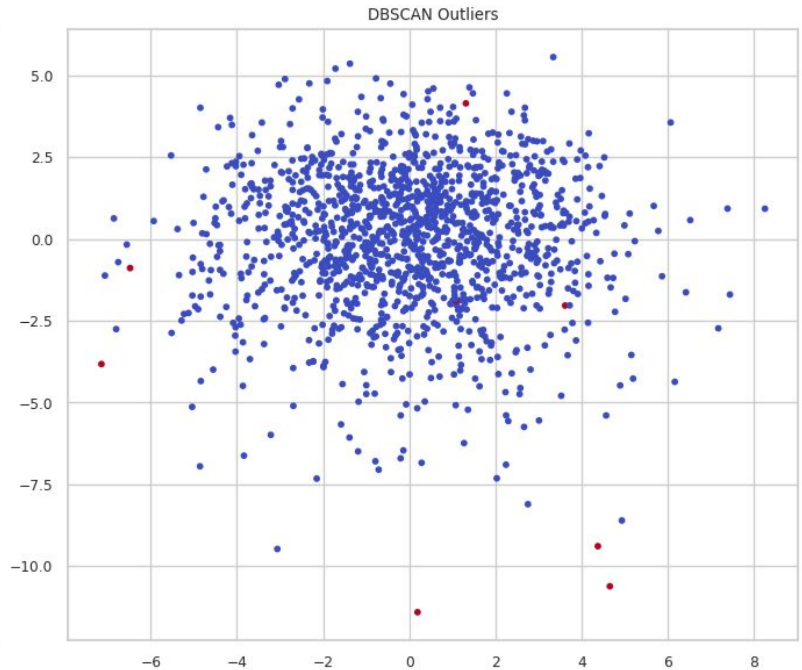
Elbow Point Analysis

- The sharpest increase in distance (the **“elbow”**) occurs at **ε ≈ 4.7**.
- This marks the transition from dense regions to isolated points.
- DBSCAN will:
  - Capture **dense clusters** with inter-point distances < 4.7.
  - Flag **isolated points** beyond this threshold as **outliers**.



Outlier Detection Comparison





- **Isolation Forest** (left): Identifies scattered anomalies across the feature space. Suitable for high-dimensional datasets and less sensitive to clustering structure.
- **DBSCAN** (right): Flags outliers that lie outside dense clusters. Effective for detecting edge cases and low-density zones.
- **Conclusion**: Both methods offer unique perspectives. Combining results enhances the robustness of anomaly detection. Outliers were used to create outlier features and added to dataset.

Next Steps

- **Modeling**
  Train and compare multiple classification models: Logistic Regression, Random Forest, XGBoost, and SVM.
- **Evaluation**
  Assess performance using Accuracy, Precision, Recall, F1-Score, and Cross-Validation.
- **Feature Importance**
  Use model-based and SHAP analyses to interpret which features drive predictions.

Project 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, XGBoost, SVM.Train RF, Logistic Regression
- **28 Apr.** Final report & presentation.

Conclusion

- The dataset was scaled, engineered with domain-specific ratios, reduced using PC, and outliers were detected to improve model performance.
- Classification models (Logistic Regression, Random Forest, XGBoost, SVM) will be used to help predict faults.
- This approach enhances model interpretability and preserves rare but important data behavior.
- The framework lays the groundwork for predictive maintenance in gas turbine operations.