2023

Predictive Maintenance for Machine Failures

FINAL REPORT

DALLEN HUANG

SPRINGBOARD | 2023-10-06

Introduction

In today's industrial landscape, minimizing unexpected machine failures is crucial for maintaining productivity and reducing operational costs. This final project report outlines our journey towards building an effective predictive maintenance model using the "AI4I 2020 Predictive Maintenance Dataset" from the UCI Machine Learning Repository. Our objective was to develop a robust binary classification model within a time series framework to predict machine failures. This report documents our approach, decisions, and results.

Data Overview

The dataset consists of 10,000 data points with 14 features, including both numerical and categorical variables. Here's a brief overview of the key columns:

- UID (Unique Identifier): A unique identifier ranging from 1 to 10,000.
- Product ID: A combination of a letter (L, M, or H) representing product quality variants and a variant-specific serial number.
- Type: Categorical variable representing product types (L, M, H).
- Air temperature [K]: Numerical variable representing air temperature in Kelvin.
- Process temperature [K]: Numerical variable representing process temperature in Kelvin.
- Rotational speed [rpm]: Numerical variable representing rotational speed in revolutions per minute.
- Torque [Nm]: Numerical variable representing torque in Newton-meters.
- Tool wear [min]: Numerical variable representing the tool wear duration in minutes.
- Machine failure: Binary variable (0 or 1) indicating whether the machine has failed for any of the failure modes.
- TWF (Tool Wear Failure): Binary variable indicating tool wear failure.
- HDF (Heat Dissipation Failure): Binary variable indicating heat dissipation failure.
- PWF (Power Failure): Binary variable indicating power failure.
- OSF (Overstrain Failure): Binary variable indicating overstrain failure.
- RNF (Random Failures): Binary variable indicating random failures.

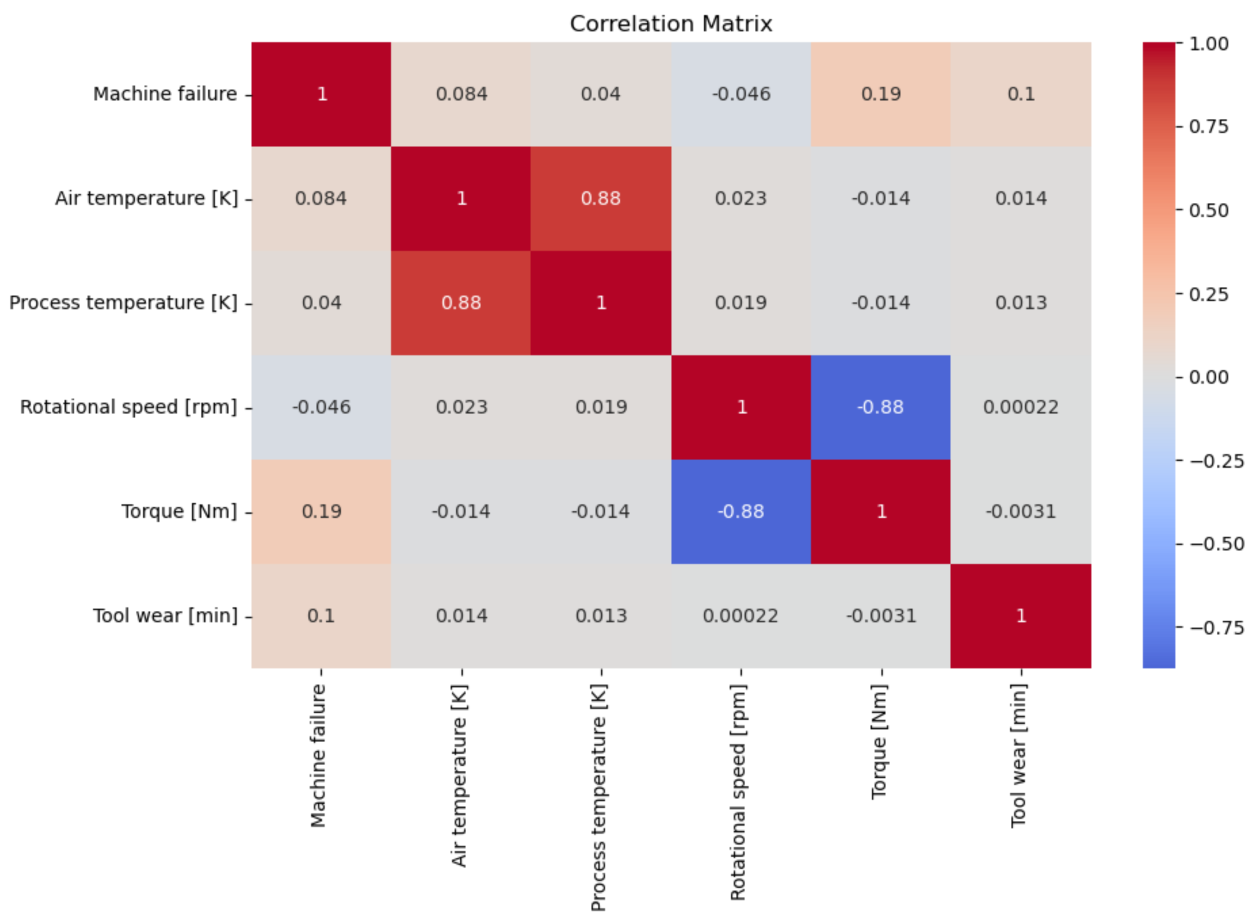
Exploratory Data Analysis

Adjusting Machine Failure and Random Failures

- The relationship between "Machine Failure" and the five single failure modes (TWF, HDF, PWF, OSF, RNF) was reviewed.
- Instances where "Machine Failure" was 1 but all five single failure modes were 0 were updated to set "RNF" (Random Failures) to 1 to align with the data description.
- Instances where at least one single failure mode was 1 but "Machine Failure" was 0 were updated to set "Machine Failure" to 1.
- The total number of failures per "Machine Failure" status was recalculated to ensure consistency.

Correlation Analysis

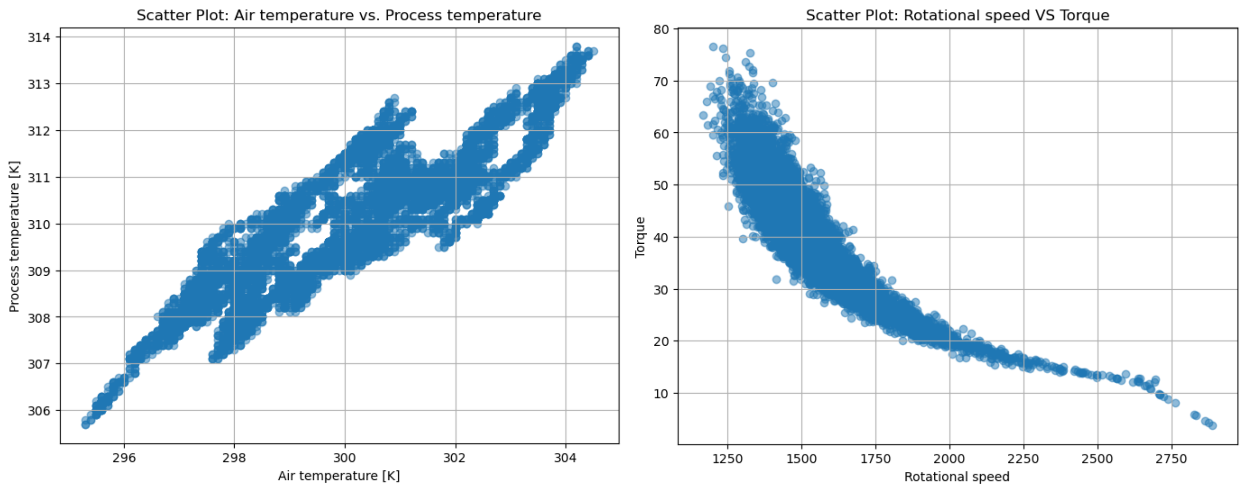
As shown in Picture 1, strong correlations were observed with the Correlation Matrix, particularly between 'Air temperature' and 'Process temperature', as well as between 'Rotational speed' and 'Torque'. These correlations align with the physical relationships we'd expect. When the process temperature increases, the air temperature tends to rise as well. Similarly, higher rotational speed typically corresponds to increased torque.



Picture 1. Correlation Matrix of Numerical Variables

The following Picture 2 reveals that 'Process temperature' and 'Air temperature' have a strong liner relationship and 'Process temperature' consistently exceeds 'Air temperature' across all machine records. This suggests that changes in 'Process temperature' drive the variations observed in 'Air temperature'. To address multicollinearity, I have chosen 'Process temperature' as an independent variable, while dropping 'Air temperature'.

'Rotational speed [rpm]' and 'Torque [Nm]' are also closely related, but the relationship is a bit more complicated than the temperature features. I've chosen to include 'Rotational speed [rpm]' as a feature for our model and decided to leave out 'Torque [Nm]'.



Picture 2. Strong Correlated Variables

Dependent and Independent Variables

Dependent variable

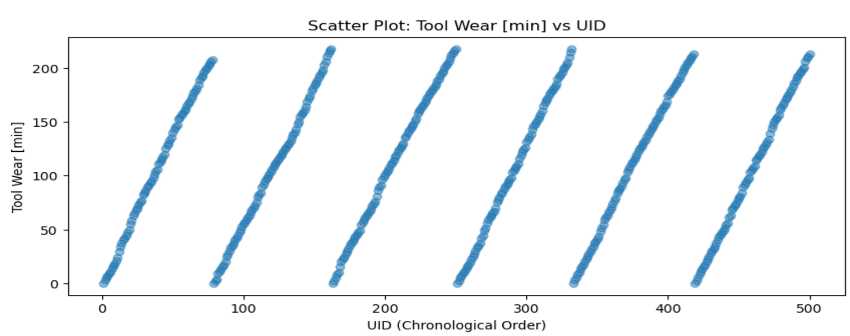
- Machine failure: representing overall machine failure status (True or False). It indicates the occurrence of at least one of the five single failure modes. There are 357 failures among 10,000 records.

Independent Variables

- UID: Ranges from 1 to 10000 and serves as the chronological order for the produced products.
- Type: Product quality variant (L, M, or H)
- Process temperature [K]: Process temperature in Kelvin
- Rotational speed [rpm]
- Tool wear [min]

Data Visualization

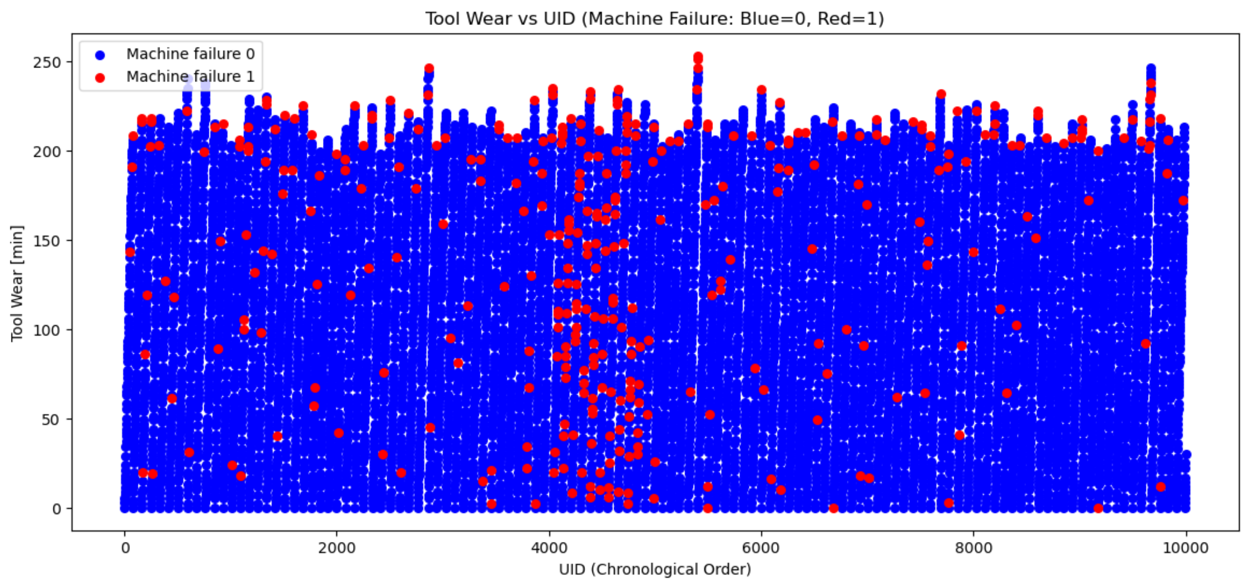
a. Tool Wear and UID (first 500 data points)



Picture 3. Tool Wear vs UID (first 500 data points)

After analyzing the Picture 3, plot of the first 500 data points, it becomes evident that 'Tool wear [min]' is periodically reset to '0' when it reaches approximately 220 minutes. This pattern suggests that the tool is changed after a specific usage time, around 220 minutes.

b. Tool Wear and UID (all 10000 data points)



Picture 4. Tool Wear vs UID (all 10000 data points)

From Picture 4, we can see two important things:

More Failures Before Changing Tools: There are more machine failures before the tools are replaced. This tells us that machines tend to break down more often just before it's time to change the tool. This is an essential pattern to consider when planning maintenance to avoid these failures.

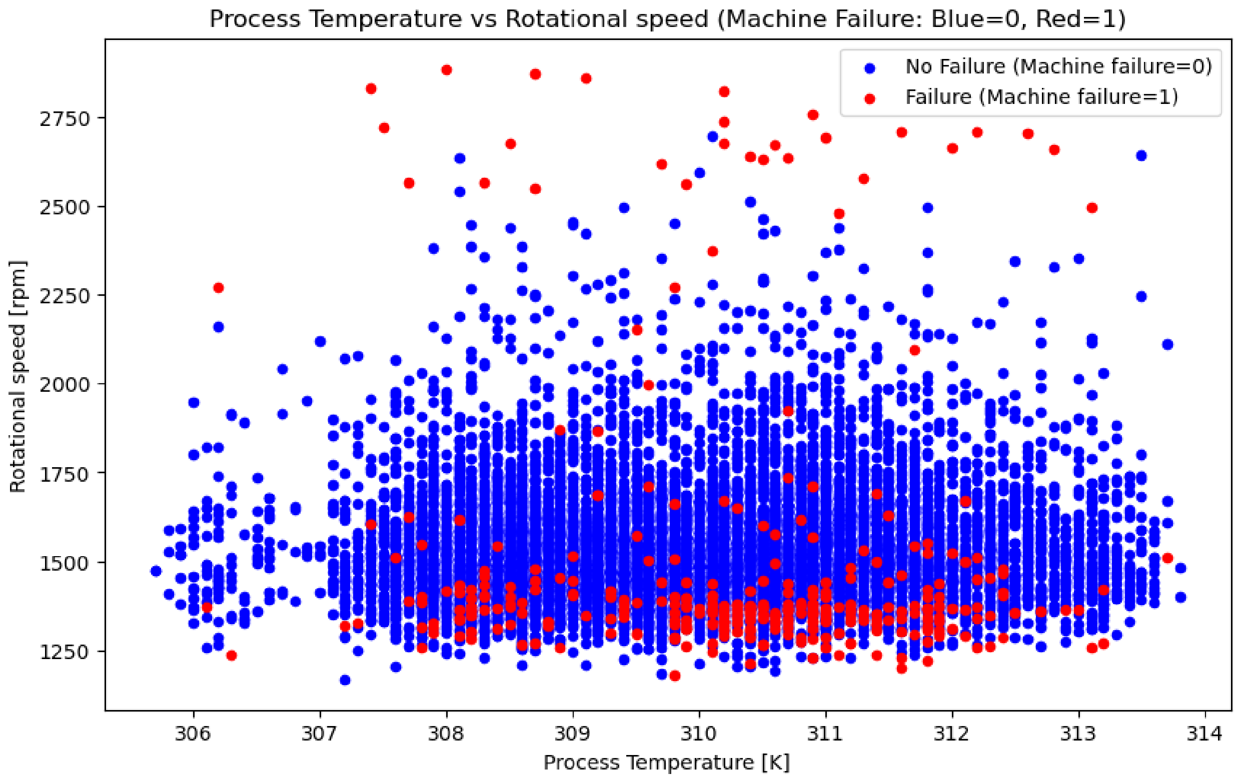
Failures During UDI 4000 to 5000: We also notice that there's a higher number of failures between UDI 4000 and 5000. UDI is like a count of the products made, and this range indicates that machines are more likely to fail during this part of production. This gives us a hint about when to pay extra attention to prevent breakdowns.

c. Process Temperature and Rotational Speed

From Picture 5, we can see two important things:

Frequent Failures at Rotational Speeds (1250 to 1500): The plot shows that the machine fails more often when the speed it's spinning at is between 1250 and 1500. This suggests that something about this speed range might be causing the failures. It's crucial to understand why this happens to find ways to prevent these failures.

Very High Failures at High Rotational Speeds (> 2000): We also notice that when the machine spins faster than 2000, the failures happen a lot more. This is a critical point because it means that when the machine goes beyond this speed, there's a significant risk of it breaking down. Investigating why this happens at higher speeds is essential to make sure the machine operates safely and reliably.



Picture 5. Process Temperature and Rotational Speed

Rescaling Features

- The 'Tool wear [min]' feature contains values ranging from 0 to over 250 minutes. Since our project is all about predicting maintenance events over time, just normalizing these values might not make much sense. Instead, the values are converted from minutes to hours. This will give us a more meaningful scale, 0 to 4+ hours, while keeping the time-related aspect intact.
- 'Process temperature [K]' and 'Rotational speed [rpm]' features are scared to a range between 0 and 1. This normalization is initially applied to the training set and then carried over to both the training and testing datasets.
- ‘Type' column, as categorical data, is converted into dummy variables with OneHotEncoder.

Modelling

Data Splitting

- Training set: the initial 80% of data points
- Testing set: the remaining 20% of data points

Machine Learning Models

- Logistic Regression

- Random Forest
- Gradient Boosting Classifier
- Long Short-Term Memory (LSTM)

Model Evaluation

- Accuracy: The proportion of correctly predicted instances.
- Precision: The ratio of true positive predictions to the total positive predictions.
- Recall: The ratio of true positive predictions to the total actual positives.
- F1-score: The harmonic means of precision and recall, balancing both metrics.
- ROC AUC (Receiver Operating Characteristic Area Under the Curve): A measure of a model's ability to distinguish between classes.

Evaluation Metrics

Model	Logistic Regression	Random Forest	Gradient Boosting Classifier	Long Short-Term Memory (LSTM)
Accuracy	98.05%	98.05%	98.10%	98.05%
Precision (Class 1)	0	0.5	0.53	0
Recall (Class 1)	0	0.23	0.26	0
F1-score (Class 1)	0	0.32	0.34	0
ROC AUC Score	0.84	0.87	0.931	0.839

Machine Learning Model Evaluation Metrics

- Logistic Regression achieves high accuracy but performs poorly in terms of precision, recall, and F1-score for Class 1 (machine failures). It fails to effectively identify machine failures, as indicated by the low precision and recall.
- Random Forest also achieves high accuracy and provides a better precision-recall balance compared to Logistic Regression. However, the recall for Class 1 is still relatively low, indicating room for improvement in identifying machine failures.
- The Gradient Boosting Classifier with default parameters shows improved performance compared to both Logistic Regression and Random Forest. It achieves the highest ROC AUC score, indicating good overall model performance. However, the recall for Class 1 remains relatively low, suggesting that there is still room for improvement in identifying machine failures.
- LSTM, a deep learning model, performs similarly to Logistic Regression in terms of accuracy and performs poorly in precision, recall, and F1-score for Class 1. It fails to effectively identify machine failures.

After comparing performance metrics of the four models, Gradient Boosting Classifier outperformed Logistic Regression, Random Forest and LSTM in terms of precision, recall, and F1-score, especially for class 1. Grid Search will be applied on Gradient Boosting Classifier to find the best hyperparameters.

Grid Search for Gradient Boosting Classifier

After comparing performance metrics of the four models, Gradient Boosting Classifier outperformed Logistic Regression, Random Forest and LSTM in terms of precision, recall, and F1-score, especially for class 1. Grid Search will be applied on Gradient Boosting Classifier to find the best hyperparameters.

We initiated the process by defining a hyperparameter grid for the GBM model. The hyperparameters considered included:

- Number of Estimators (n_estimators): [50, 100, 200]
- Learning Rate (learning_rate): [0.01, 0.1, 0.2]
- Maximum Depth (max_depth): [3, 4, 5]
- Subsample (subsample): [0.8, 0.9, 1.0]

We created a GridSearchCV object to systematically search through this parameter grid while performing 5-fold cross-validation. The performance metric used for evaluation was the ROC AUC score.

The best hyperparameters identified by the grid search were as follows:

- Learning Rate: 0.01
- Maximum Depth: 3
- Number of Estimators: 200
- Subsample: 1.0

Cross-Validation

With the best hyperparameters, we proceeded to evaluate the model's performance using cross-validation. We employed 5-fold cross-validation with StratifiedKFold, which is particularly useful for datasets with imbalanced classes. The purpose of cross-validation is to estimate how well the model is likely to perform on unseen data.

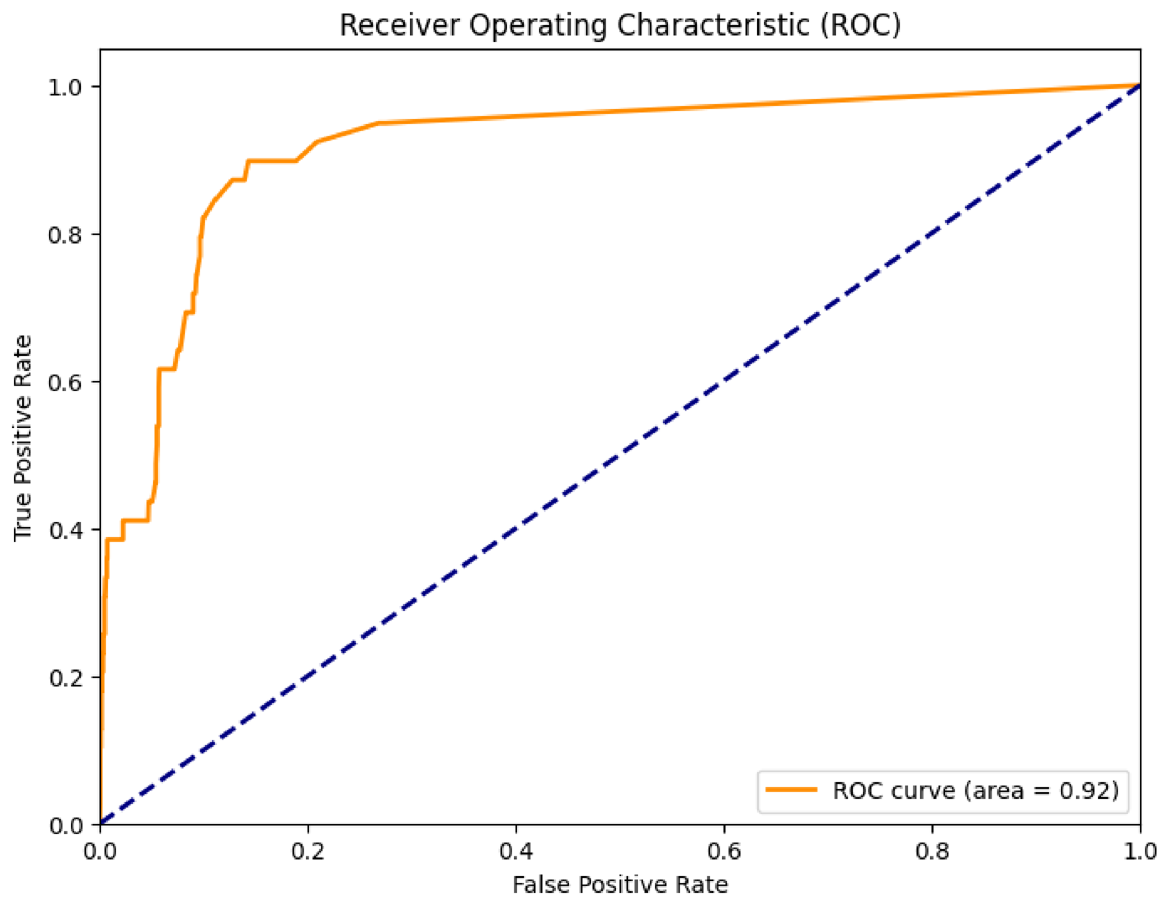
The cross-validation results demonstrated consistent accuracy scores across folds:

- Cross-Validation Accuracy Scores: [0.9625, 0.959375, 0.9675, 0.9625, 0.9625]
- Mean Accuracy: 0.962875
- Standard Deviation of Accuracy: 0.002610076627227651

Final Model Evaluation

Finally, we assessed the GBM model's performance on the testing dataset with the optimized hyperparameters. The results on the testing dataset were as follows:

- Accuracy: 0.9815
- Precision (Class 1): 0.57
- Recall (Class 1): 0.21
- F1-score (Class 1): 0.30
- ROC AUC Score: 0.9158919441938309



Picture 6. ROC of Final Model

Conclusion and Future Work

This project demonstrated a systematic approach to predictive maintenance using a variety of machine learning techniques. The Gradient Boosting Classifier, after hyperparameter tuning, emerged as the best-performing model, achieving a remarkable accuracy of 98.15% and a robust ROC AUC score of 0.916. However, precision and recall for class 1 were areas for potential improvement.

Further Research

- Study How Failures Spread Over Time: Look closely at four specific failure types and how they develop over time in the dataset.
- Keep Improving the Model and Try Different Ways: Work on making the model better and test other methods like changing how it weighs different results. This helps to find a good balance between being right a lot and catching as many issues as possible.
- Check How Data Is Gathered and Make It Better: Review how the data is collected. Make it more accurate and useful to make sure the maintenance predictions are as good as they can be.

Recommendations for the Client

- Implement the model in maintenance process. Use it to predict when machines are likely to fail and plan maintenance more efficiently. This can help reduce unexpected downtimes and improve overall operational efficiency.
- Continuously collect new data and retrain the model periodically. This ensures it adapts to changing conditions and remains a valuable tool for predictive maintenance.
- Examine the conditions under which machines operate at high speeds. Check for any factors like overheating or mechanical stress that might lead to failures. Implement measures to mitigate these issues and reduce high-speed failures.
- Encourage open communication between these teams. Ensure that maintenance personnel understand the model's predictions and can use them effectively. Data teams should continuously work on model improvement based on feedback from maintenance experts.

Reference

1. UCI Machine Learning Repository. (n.d.). AI4I 2020 Predictive Maintenance Dataset. Retrieved from https://archive.ics.uci.edu/ml/datasets/AI4I+2020+Predictive+Maintenance+Dataset