PREDICTIVE MAINTENANCE FOR MACHINES

A machine Learning approach to predict machine failure

BUSINESS UNDERSTANDING

Manufacturing industries rely heavily on equipment that must operate efficiently and continuously to meet production targets. Unplanned machine failures result in costly downtime, repair expenses, and lost productivity. Traditional maintenance strategies—either reactive (fix after failure) or preventive (scheduled maintenance)—often lead to inefficiencies, either by acting too late or too early.

GOALS

- Develop a binary classification model to predict machine failure (No Failure OR Failure).
- Perform Exploratory Data Analysis (EDA) to uncover key drivers of failure.
- Rank the importance of sensor features contributing to failure.
- Build a real-time or batch scoring system for predictive alerts.

WHY IT MATTERS

- Most companies currently experiences production losses due to unexpected machine failures. While sensor data is available, it is not being fully leveraged to predict when a failure is imminent.

DATA UNDERSTANDING

The dataset consists of 10,000 records and 10 columns, collected from manufacturing machines equipped with various sensors.

Important columns include:

- Type of machine
- Air temperature [K]
- Process temperature [K]
- Rotational speed [rpm]
- Torque [Nm]
- Tool wear [min]
- Binary indicator of failure — 0 for no failure, 1 for failure.

DATA PREPARATION

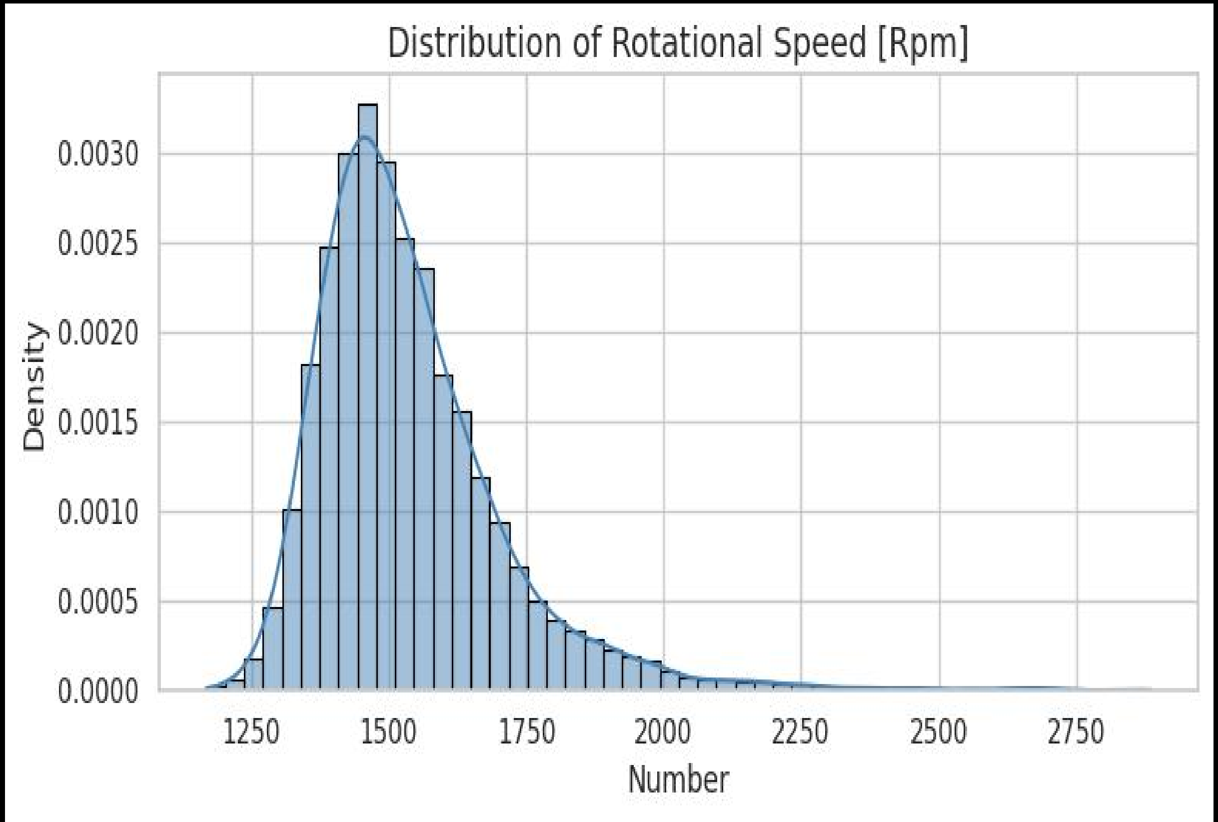
The process involved:

- Dropping of unnecessary columns such as product ID
- Adding new features such as; temperature difference
- Data preprocessing i.e label encoding and scaling of the features

EXPLORATORY DATA ANALYSIS

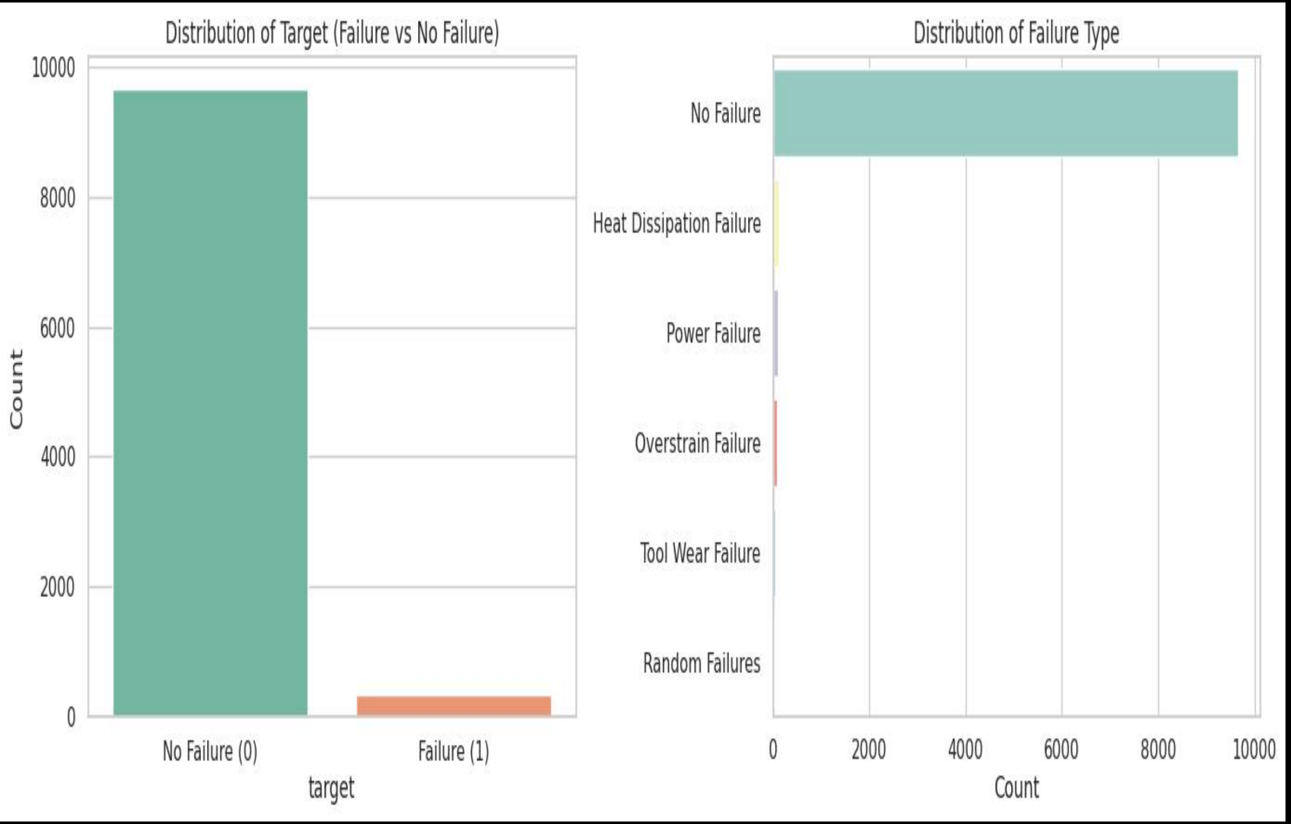
- Univariate Analysis
- Bi-variate Analysis
- Multi-variate Analysis

DISTRIBUTION OF ROTATION SPEED



- The distribution is clearly **right-skewed (positively skewed)**. This means the tail of the distribution extends further to the right.
- The peak of the distribution, which represents the most frequent rotational speeds, appears to be around **1450-1500 RPM**.

DISTRIBUTION OF THE TARGET



- The 'Distribution of Target' plot shows a significant class imbalance. The number of instances with 'No Failure' (0) is much higher than those with 'Failure' (1).

MODELING

For the modeling part build FOUR traditional models that are good with classification. These models are:

- Logistic Regression
- Random Forest Classifier
- XGBoost
- Support Vector Machine

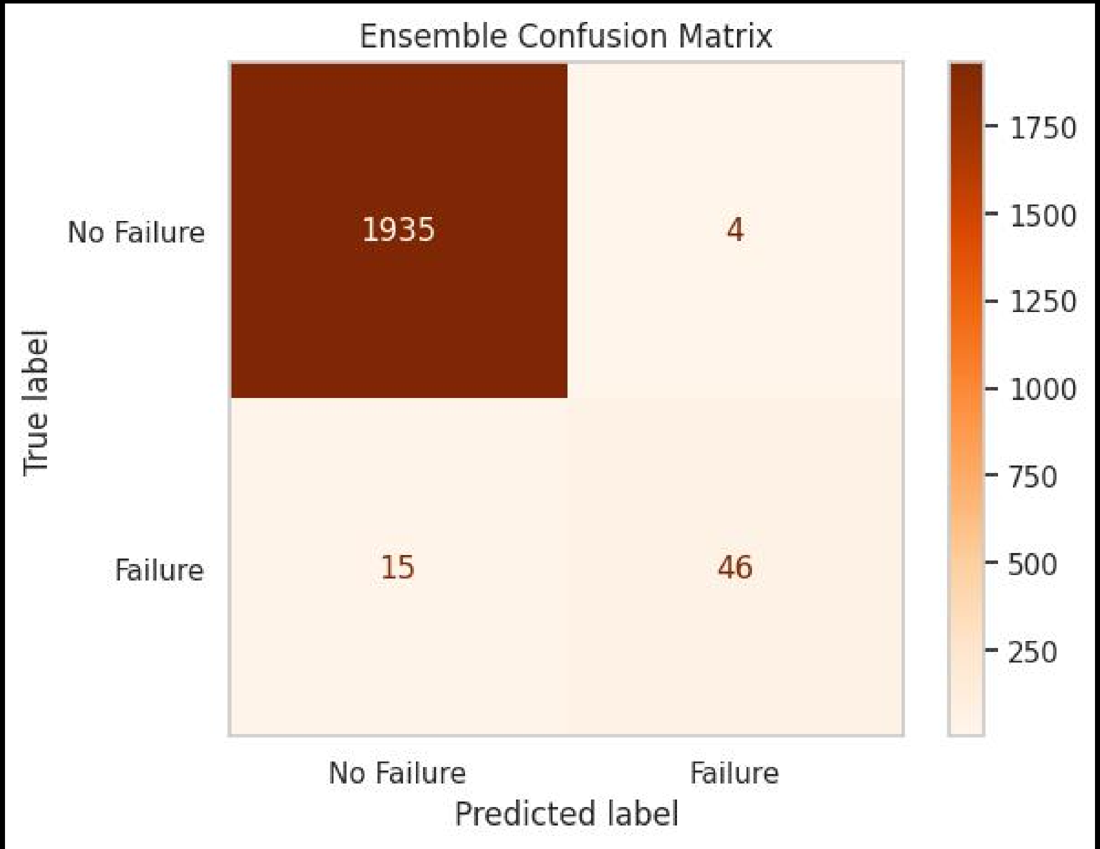
For the deep learning models tried out Artificial Neural Network (ANN)

The best performing models were the Random Forest and XGBoost which was ensembled and performed much better

EVALUATION

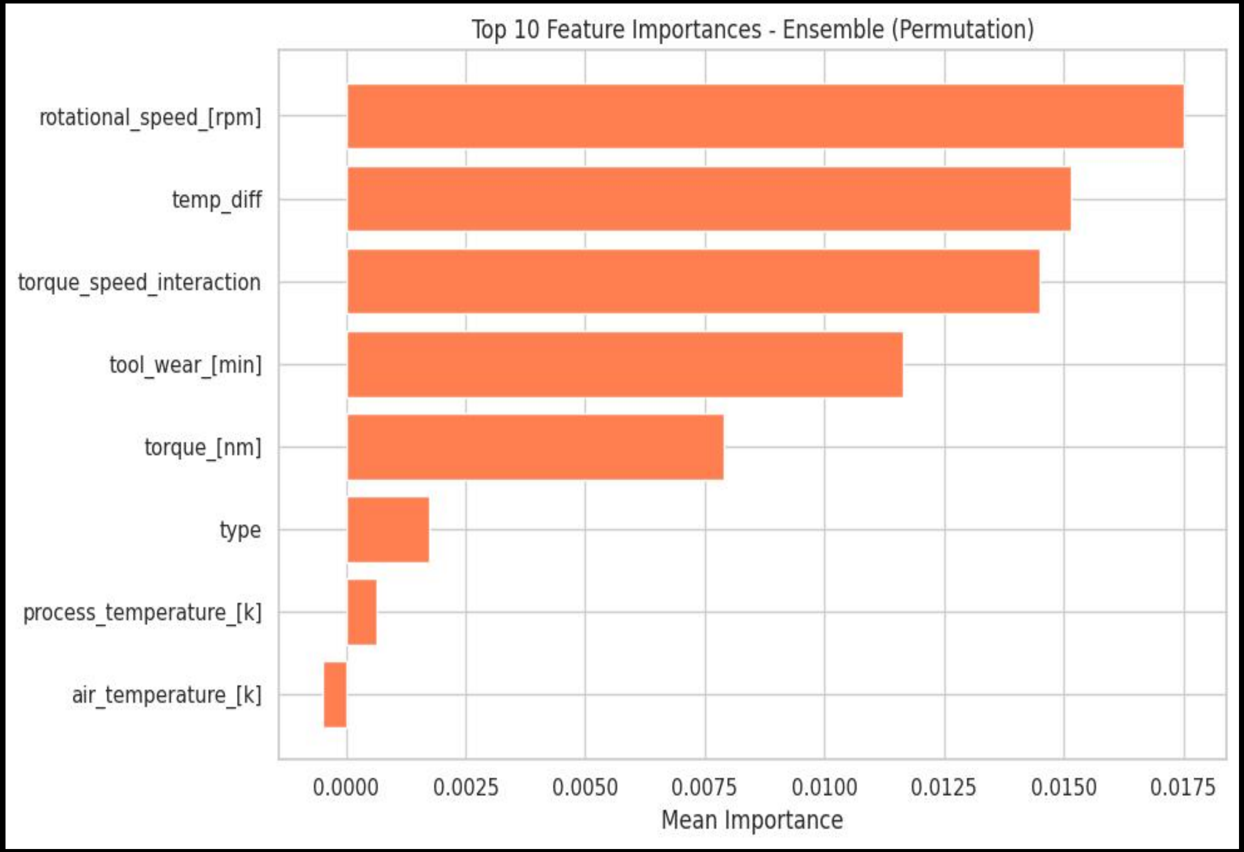
MODEL	ACCURACY	F1-SCORE
ENSEMBLE MODEL	99%	91%
RANDOM FOREST	99%	91%
XGBOOST	99%	90%
SUPPORT VECTOR MACHINE	98%	83%
LOGISTIC REGRESSION	87%	60%
NEURAL NETWORK	98%	81%

ENSEMBLE MODEL CONFUSION MATRIX



- True Negatives (No Failure correctly predicted as No Failure): 1935 instances.
- This is the highest number of correctly predicted "No Failure" instances among all models reviewed. T- rue Positives (Failure correctly predicted as Failure): 46 instances.
- False Positives (No Failure incorrectly predicted as Failure): 4 instances. This is the lowest number of false positives among all models reviewed, indicating very few instances where "No Failure" was wrongly flagged as "Failure."
- False Negatives (Failure incorrectly predicted as No Failure): 15 instances. This is tied with the Random Forest model for the lowest number of false negatives, meaning it missed fewer actual "Failure" events.

FEATURE IMPORTANCE GRAPH



Dominant Features:

- rotational_speed_[rpm] is by far the most important feature, with the highest mean importance value (around 0.0175).
- temp_diff and torque_speed_interaction are also highly important, following rotational_speed_[rpm] closely.
- tool_wear_[min] and torque_[nm] are the next most important features, though with a noticeable drop in importance compared to the top three.

CONCLUSION

- Ensemble Model Outperforms Neural Network The ensemble model delivered superior performance across key metrics, especially for the minority class (failure).
- There was effective Failure Detection with Fewer Errors
- Neural Network Struggles with Class Imbalance

RECOMMENDATIONS

- Adopt the Ensemble Model for Deployment
- Monitor for Model Drift
- Implement Model Explainability