



Best Machine Ever

Yara Yaghi, Muhammad Arfin, Shakir Azami, Paula Schultz, Austin Yoo, Sheena
Gandham

Business Problem

How can we reduce the frequency of factory machine failures by identifying and addressing the most impactful failure causes?

- Reducing the frequency of factory machine failures is crucial for the efficiency and reliability of machines (Marcellus, 2024)
- Identifying the most impactful failures is important for productivity, optimizing maintenance costs, and improving safety (Sensemore, 2024)

Background

ASSEMBLY BREAKING NEWS

Equipment Failure Is Costly for Manufacturers



August 2, 2021

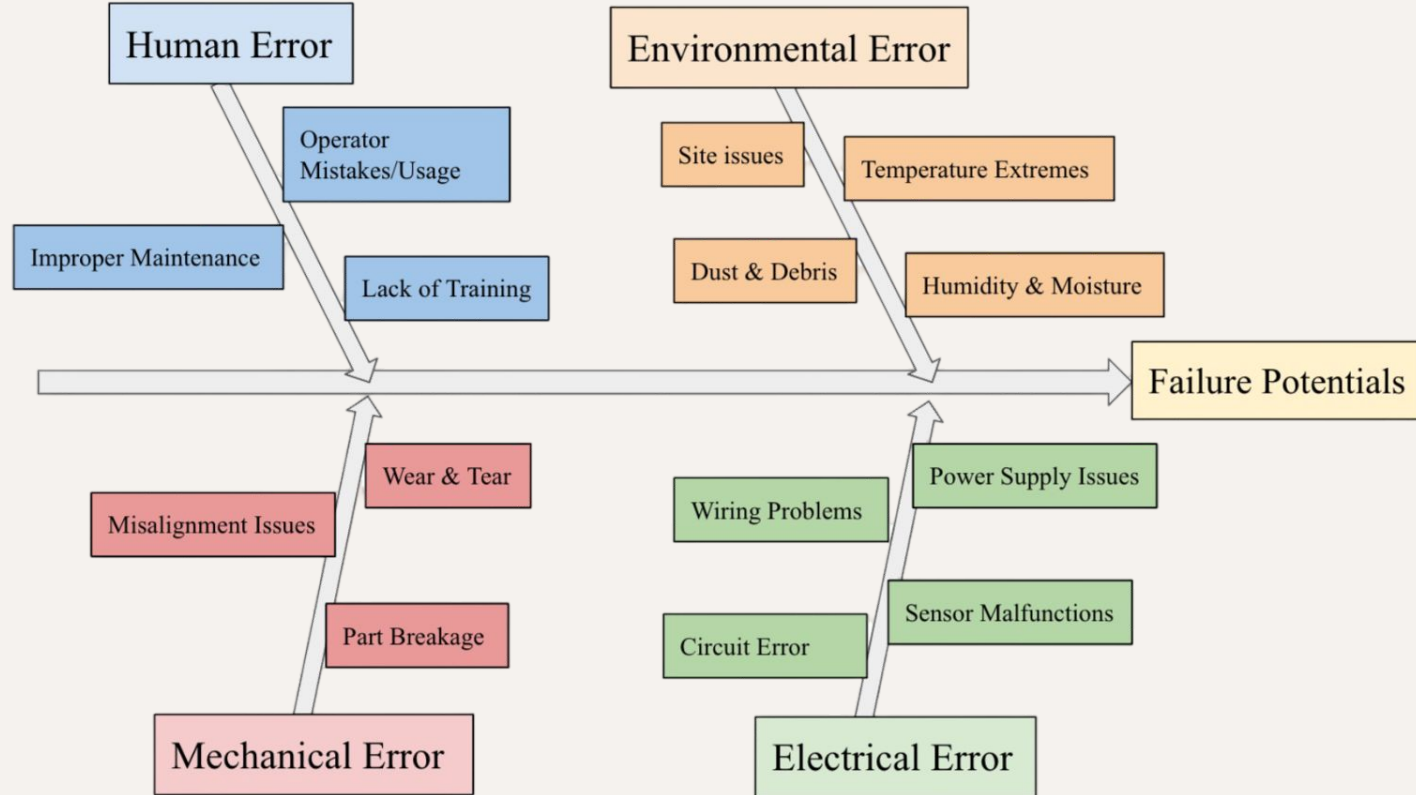


Equipment Malfunction | A Costly Business Challenge and How to Overcome It

Equipment Malfunction / Equipment Malfunction / By Marcellus

- Machine failures can result in a loss of production hours, unintended downtimes, and cause issues concerning safety (Marcellus, 2024)
- Studies estimate the average downtime cost from machine failure is up to about \$532,000 per hour (Weber et al., 2021)

Analytic



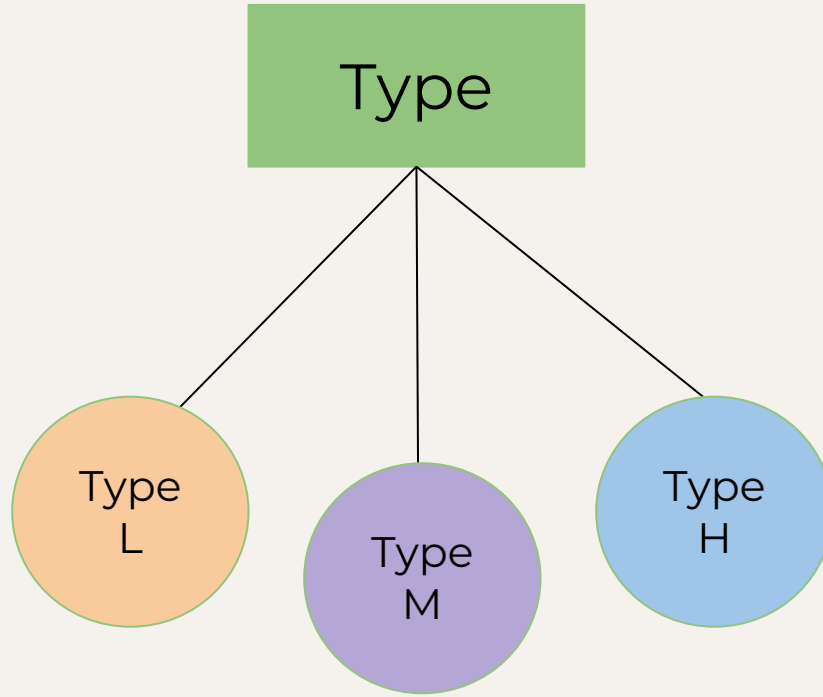
Data Summary

- **Name:** AI4I 2020 Predictive Maintenance Dataset
- **Original Data Set:** Matzka, Stephan. "Explainable Artificial Intelligence for Predictive Maintenance Applications." *2020 Third International Conference on Artificial Intelligence for Industries (AI4I)* (2020): 69-74. (MLA)
- **Data File:** Excel Spreadsheet (2D Array)
- **Structure:** Tabular
- **Rows:** 10,000 rows
- **Features:** 15
- **Target:** derived binary failure variable

Data Features

| | Feature Name | Data Type | Missing Values | Sample/Unique Values | Description |
|----|-------------------------|-----------|----------------|---|--|
| 0 | UDI | int64 | 0 | 5190 | Unique identifier for each data point |
| 1 | Product ID | object | 0 | [H35323, L47249, M16184, L48676, L50791, L5103... | ID representing the product being manufactured |
| 2 | Type | object | 0 | [H, L, M] | Category of the product (H, L, M) |
| 3 | Air temperature [K] | float64 | 0 | 298.5 | Temperature of the air in Kelvin |
| 4 | Process temperature [K] | float64 | 0 | 311.1 | Temperature of the process in Kelvin |
| 5 | Rotational speed [rpm] | int64 | 0 | 1596 | Speed of the machine in rotations per minute |
| 6 | Torque [Nm] | float64 | 0 | 72.0 | Torque applied during operation in Newton-meters |
| 7 | Tool wear [min] | int64 | 0 | 210 | Time of tool usage before wear in minutes |
| 8 | Machine failure | int64 | 0 | 0 | Binary indicator of machine failure |
| 9 | TWF | int64 | 0 | 0 | Tool wear failure indicator |
| 10 | HDF | int64 | 0 | 0 | Heat dissipation failure indicator |
| 11 | PWF | int64 | 0 | 0 | Power failure indicator |
| 12 | OSF | int64 | 0 | 0 | Overstrain failure indicator |
| 13 | RNF | int64 | 0 | 0 | Random failure indicator |

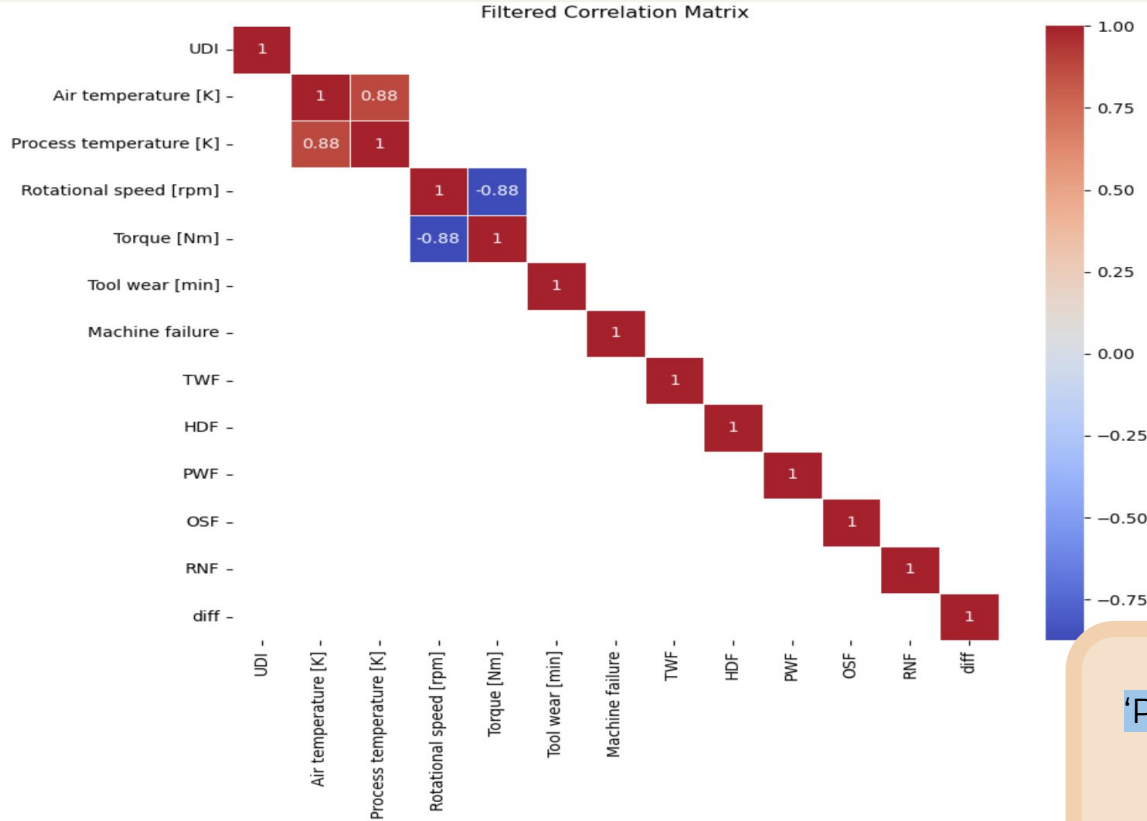
One Hot Encoding



- Processed machine type column (H/M/L) using one-hot encoding
- Updated failures to boolean

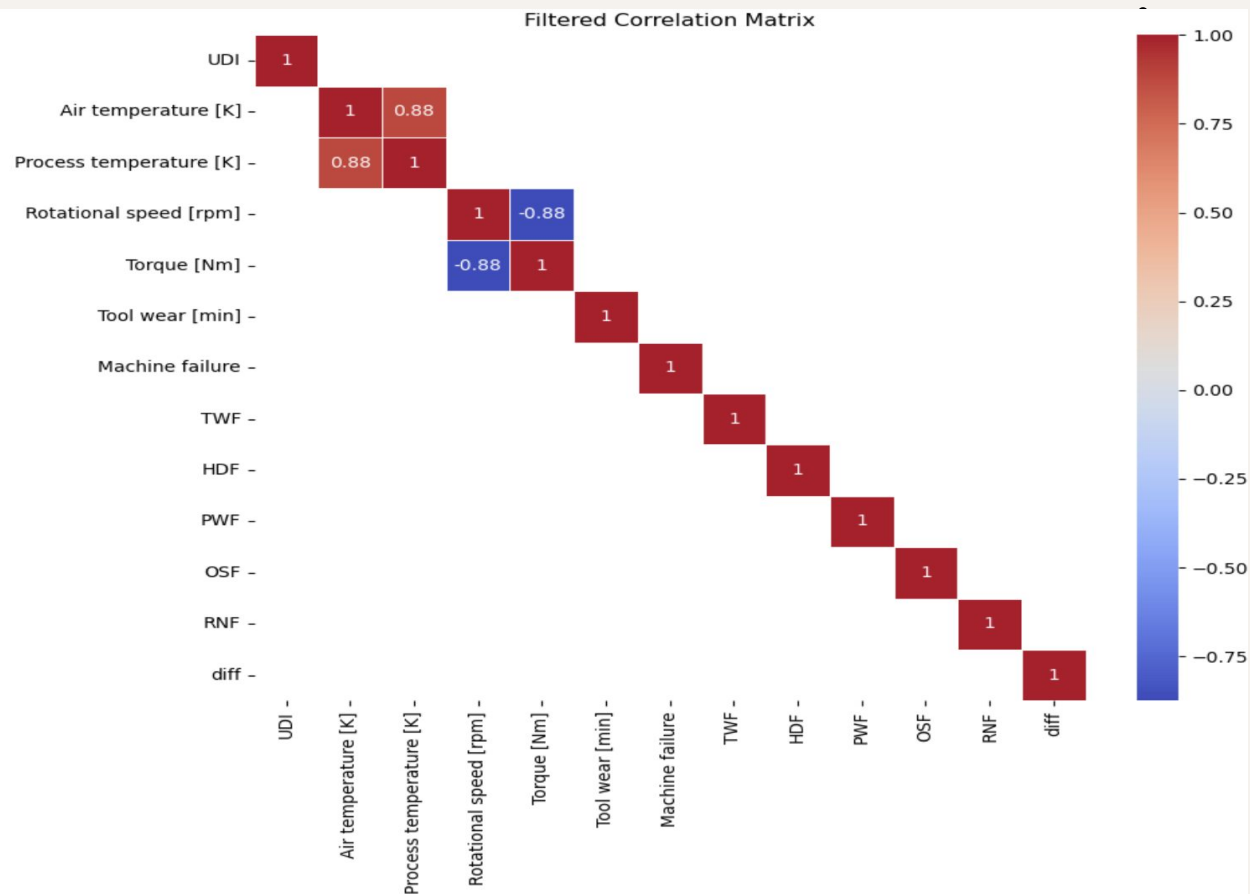
Type means Quality
(Low, Medium, High)
(2/3/5 mins to cause failure)

Feature Selection



High relationships between
 “Air temperature”
 “Process temperature”
 And
 “Rotational speed”
 “Torque”.
 Merged into
 “Temperature difference”
 “Power”
 “Tool wear [min]” adjusted
 to account for quality

Temperature Difference =
 ‘Process Temperature’ - ‘Air Temperature’
 Power = ‘Rotational speed’ * ‘Torque’
 Tool Wear Adjust =
 H/M/L - 5/3/2 min from Tool Wear



Dropped Features:
UDI
ProductID

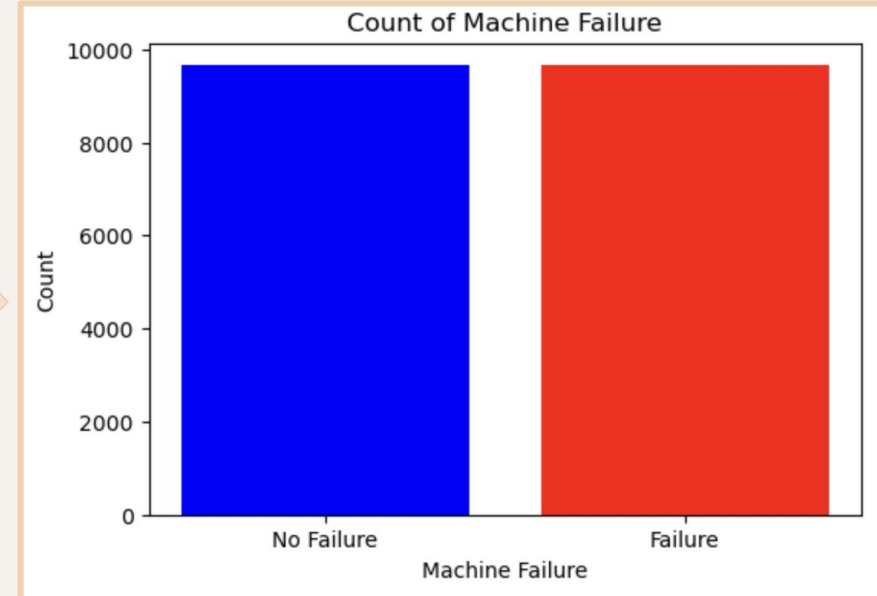
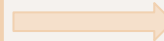
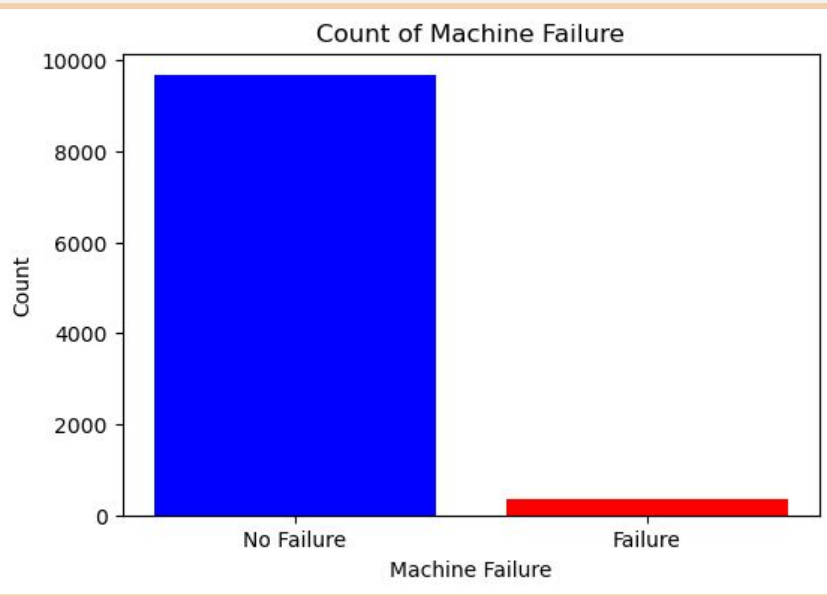
Scaling

| | Air temperature [K] | Process temperature [K] | Rotational speed [rpm] | Torque [Nm] | Tool wear [min] |
|--------------|---------------------------|-------------------------------|---------------------------|--------------|--------------------|
| count | 10000.000000 | 10000.000000 | 10000.000000 | 10000.000000 | 10000.000000 |
| mean | 300.004930 | 310.005560 | 1538.776100 | 39.986910 | 107.951000 |
| std | 2.000259 | 1.483734 | 179.284096 | 9.968934 | 63.654147 |
| min | 295.300000 | 305.700000 | 1168.000000 | 3.800000 | 0.000000 |
| 25% | 298.300000 | 308.800000 | 1423.000000 | 33.200000 | 53.000000 |
| 50% | 300.100000 | 310.100000 | 1503.000000 | 40.100000 | 108.000000 |
| 75% | 301.500000 | 311.100000 | 1612.000000 | 46.800000 | 162.000000 |
| max | 304.500000 | 313.800000 | 2886.000000 | 76.600000 | 253.000000 |

| | norm_power | norm_temp_diff | norm_tool_wear_adjusted |
|--------------|--------------|----------------|-------------------------|
| count | 19304.000000 | 19304.000000 | 19304.000000 |
| mean | 0.600788 | 0.470674 | 0.494346 |
| std | 0.168741 | 0.229215 | 0.255698 |
| min | 0.000000 | 0.000000 | 0.000000 |
| 25% | 0.501356 | 0.299809 | 0.278431 |
| 50% | 0.606479 | 0.444444 | 0.505882 |
| 75% | 0.715278 | 0.672361 | 0.729412 |
| max | 1.000000 | 1.000000 | 1.000000 |

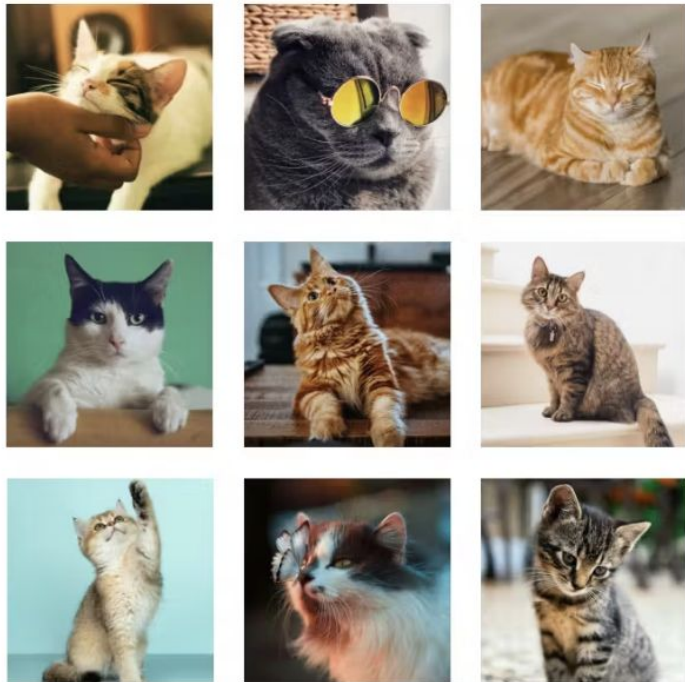
Balanced

Synthetic Minority Over-sampling Technique (SMOTE) to balance dataset.



Oversampled

Cats



Dogs



Assumptions

Assumptions Before Cleaning Dataset:

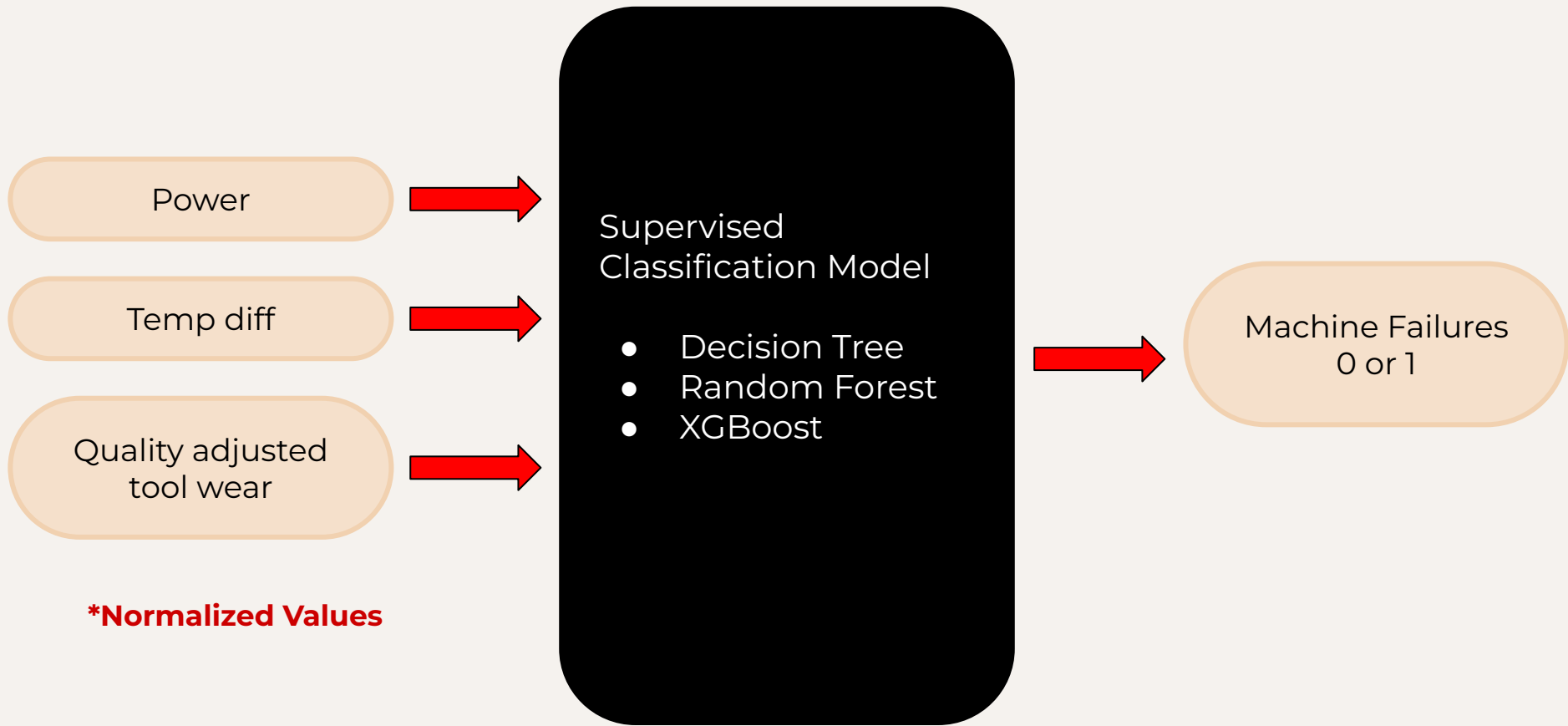
- The 'Machine Failure' labels (0 and 1) are accurate.
- Sensor data reflects actual machine health.
- Each row is treated as an independent observation.

Assumptions After Cleaning Dataset:

- If any subsystem fails, we count it as a full machine failure.
- New Features created improve the model's predictive power.
- SMOTE-generated samples are representative enough to balance the dataset.
- Normalizing the data maintains important relationships.

| # | Article/Date | Author | Method Used | Improvements to our work |
|---|--|---------------------|--|---|
| 1 | "Explainable Artificial Intelligence for Predictive Maintenance Applications." (2020) | Stephan Matzka | Supervised Classification (Decision Tree, Random Forest) | Focusing on the root cause analysis of failure rather than just predicting failure events. |
| 2 | "Predicting Machine Failures from Multivariate Time-Series: An Industrial Case Study" (2024) | Nicoló Vago et al. | Time-Series Analysis | Integrates machine learning algorithms for predicting and explaining failure causes, and real-time monitoring is considered for dynamic insights. |
| 3 | "Causes and Impact of Human Error in Maintenance of Mechanical Systems" (2020) | Mfundo Nkosi et al. | Human Factors Analysis | Focusing on predictive maintenance using data analytics and machine learning algorithms to predict potential failures before they occur. |

| # | Article/Date | Author | Method Used | Improvements to our work |
|---|---|-----------------------|--|--|
| 4 | “Towards prediction of machine failures: overview and first attempt on specific automotive industry application” (2020) | Vincent Ciano et al. | Predictive Health Monitoring (PHM), FMEA, Bayesian Networks, statistical models, and AI-driven techniques. | They use traditional predictive maintenance with regression model and failure mode analysis based on expert input which doesn't account for complex interactions between failure causes. Our approach will improve predictive accuracy, identify key failure causes and provide more interpretable results for cost effective maintenance. |
| 5 | “Predicting machine failures using machine learning and deep learning algorithms” (2024) | Devendra Yadav et al. | Machine Learning and Deep Learning | Focuses on comparing ML models (Random Forest, XGBoost, etc.) with deep learning (LSTM) to predict machine failures. It highlights hyperparameter optimization for better performance. |



Algorithm Selection for Failure Prediction

- Problem: Classification to Predict Failure from Features
- Selected Algorithms:
 - Decision Tree (DT): Capture non-linearities, may overfit
 - Random Forest: Ensemble of DT, may reduce overfit
 - XG Boost: Gradient Boosted Trees, high performance and accuracy, can utilize GPU support
 - Stacking Ensemble: Combine predictions from other models to potentially improve performance
 - MLP and KNN are rejected not handle collinearity

Hyperparameter Optimization

Configuration settings to improve performance and reduce overfitting

Not part of original data, tuned iteratively in model development

Ranges Tuned to control complexity and reduce overfitting and keeping accuracy

| Decision Tree: | Random Forest: | XGBoost: |
|--|---|--|
| GridSearchCV Exhaustive Search over Values | RandomizedSearchCV Efficient Random Search | RandomizedSearchCV and RepeatedKFold for Robustness |
| criterion: ['gini', 'entropy'] max_depth: [6, 8, 10] min_samples_split: [5, 10, 15] min_samples_leaf: [2, 4, 5, 7] ccp_alpha: [0.0, 0.001, 0.01] | n_estimators: 50-200 max_depth: 3-20 min_samples_split: 2-10 min_samples_leaf: 1-4 | n_estimators: 100-300 max_depth: 4-10 learning_rate: 0.01-0.15 subsample/colsample_bytree: 0.7-1.0 gamma/reg_alpha/reg_lambda: 0-1 / 0-1 / 1-3 (Regularization) |

| Model | Final Hyperparameters after Tuning |
|----------------|--|
| DT (Base) | Basic decision tree with default parameters. |
| DT (Best) | Tuned with GridSearchCV: 'ccp_alpha': 0.0, 'criterion': 'gini', 'max_depth': 10, 'min_samples_leaf': 2, 'min_samples_split': 5 |
| RF (Base) | Basic random forest with default parameters. |
| RF (Best) | Tuned with RandomizedSearchCV: 'max_depth': 18, 'min_samples_leaf': 1, 'min_samples_split': 3, 'n_estimators': 160 |
| XGBoost (Base) | Baseline classifier on GPU with DMatrix & early stopping. |
| XGBoost (Best) | Tuned using RandomizedSearchCV, refitted with DMatrix. 'subsample': 0.9, 'reg_lambda': 1.5, 'reg_alpha': 0.1, 'n_estimators': 200, 'max_depth': 10, 'learning_rate': 0.1, 'gamma': 0.5, 'colsample_bytree': 0.9} |

Metrics

1. Accuracy, most popular evaluation metrics for classification model.

- * balanced data 50% failure and 50%non failure

$$\text{acc} = \text{tp} + \text{tn} / \text{tp} + \text{tn} + \text{fp} + \text{fn}$$

- * using decision tree, random forest and XG boost

$$\text{acc} = 1759 + 1644 / 3861 = 0.88 \%$$

- * large data/ classification

$$\text{Rec} = \text{tp} / \text{tp} + \text{fn}$$

2. Recall, most failures can be detected

$$\text{Rec} = 1759 / 1759 + 217 = 0.89 \%$$

1. Using classification

$$\text{pre} = \text{tp} / \text{tp} + \text{fp}, \text{pre} = 1759 / 2000 = 0.87$$

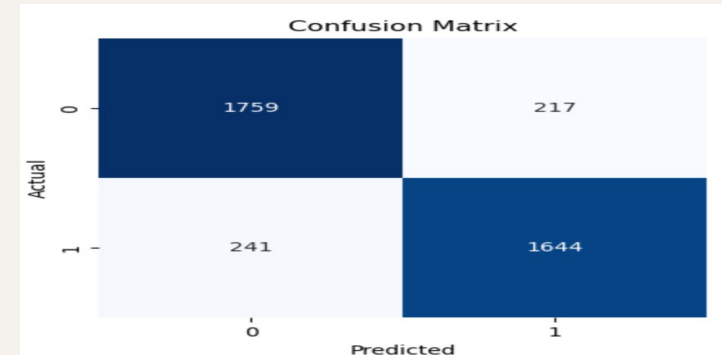
2. number failure, missing failures

3. Precision, predicts actual number of failures

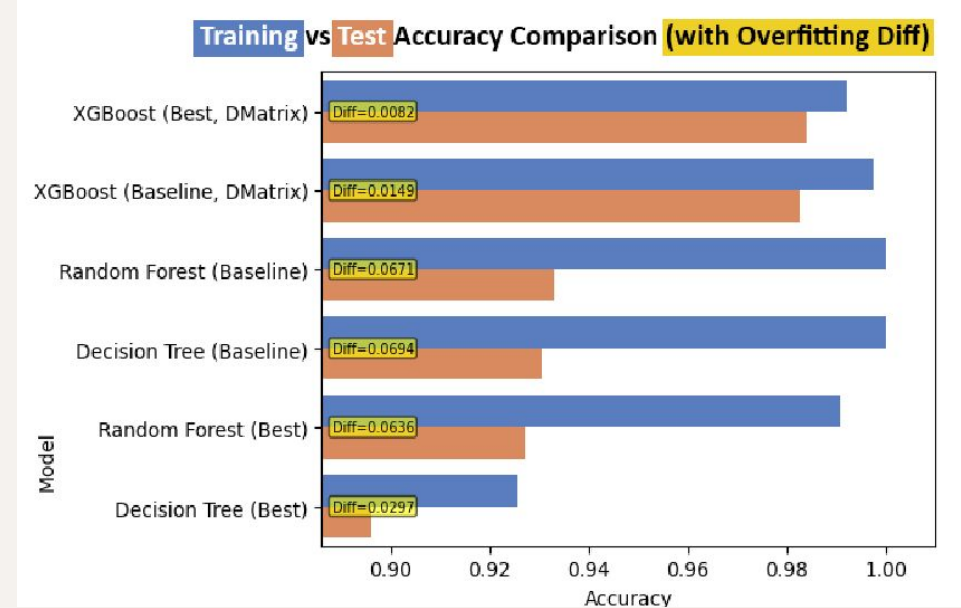
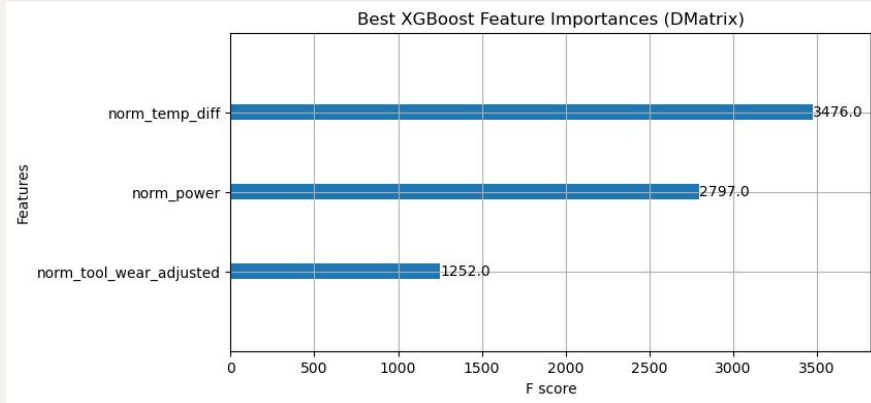
- * predicted failure vs actual failures

- * high precision means lower false positive

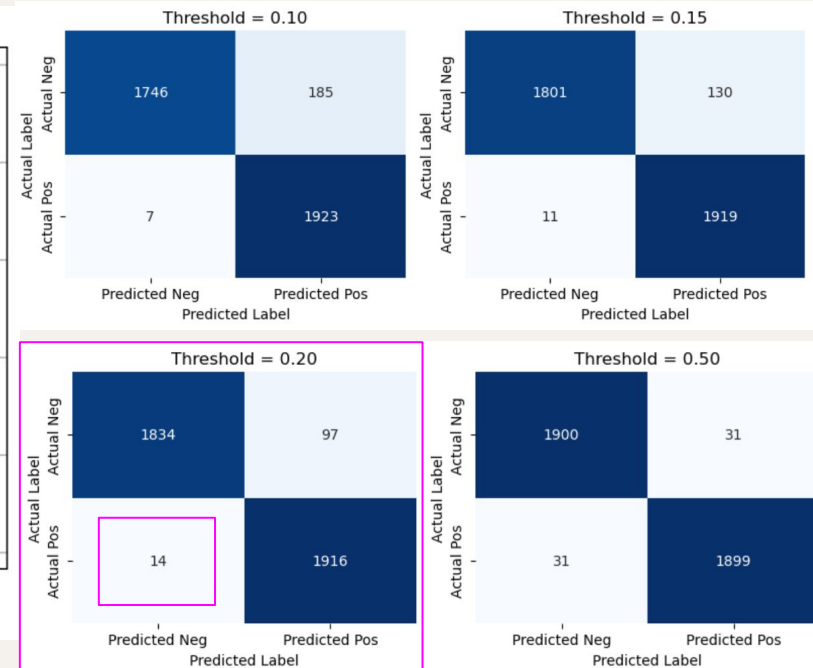
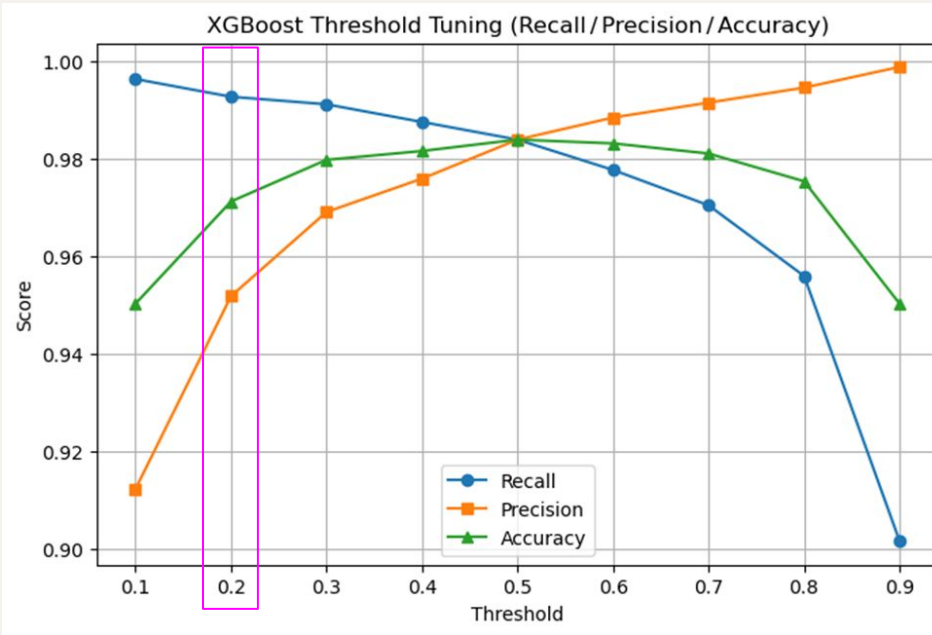
- * machine failure prevents unnecessary cost



Initial Results

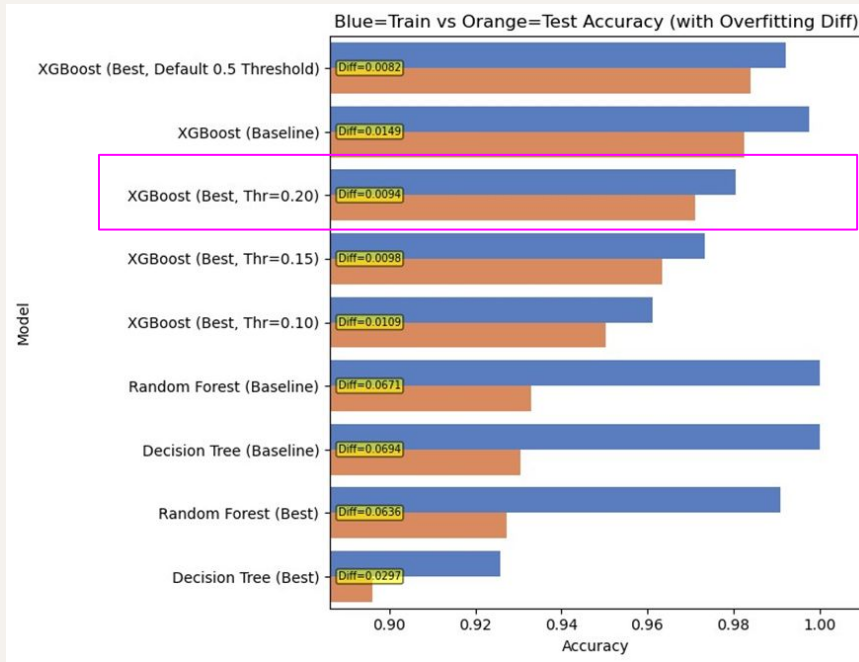


Final Results - Threshold Adjustment



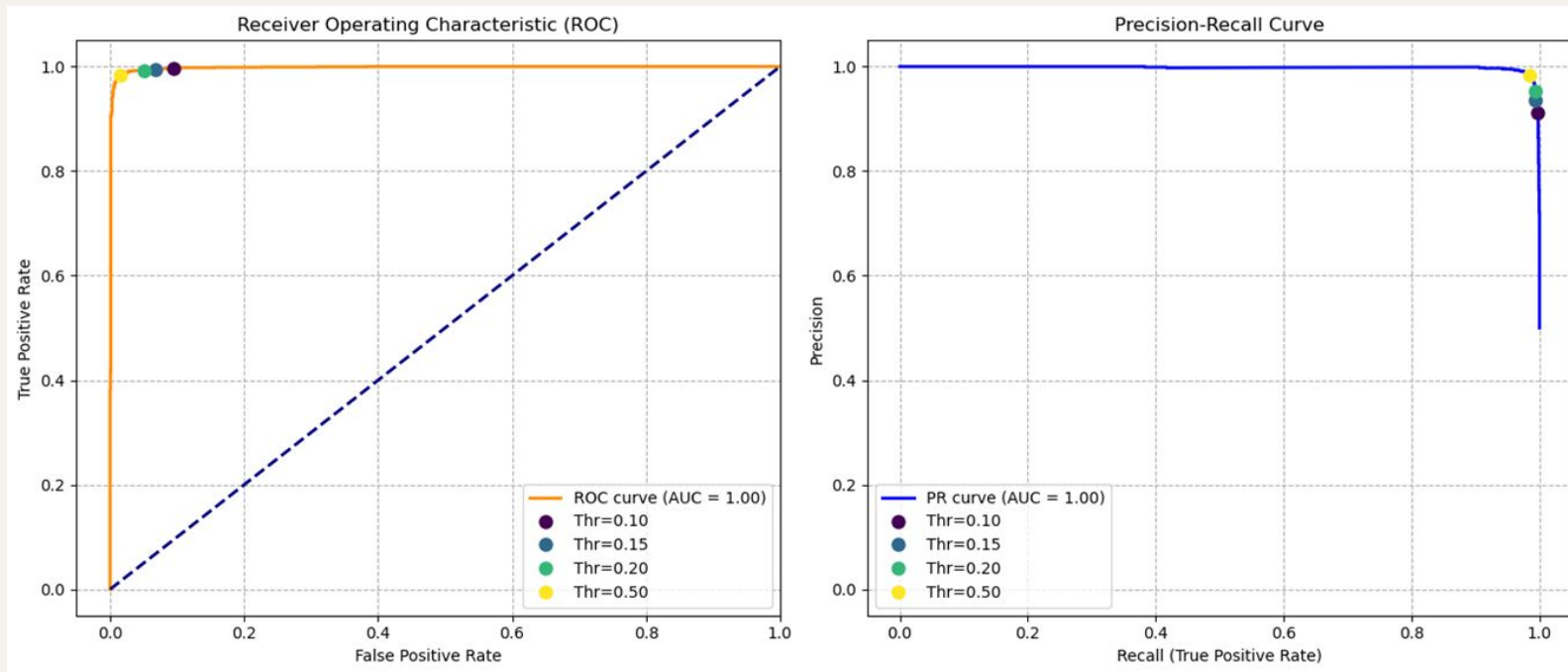
Lower threshold, fewer expensive false negatives

Final Results - Threshold Adjustment



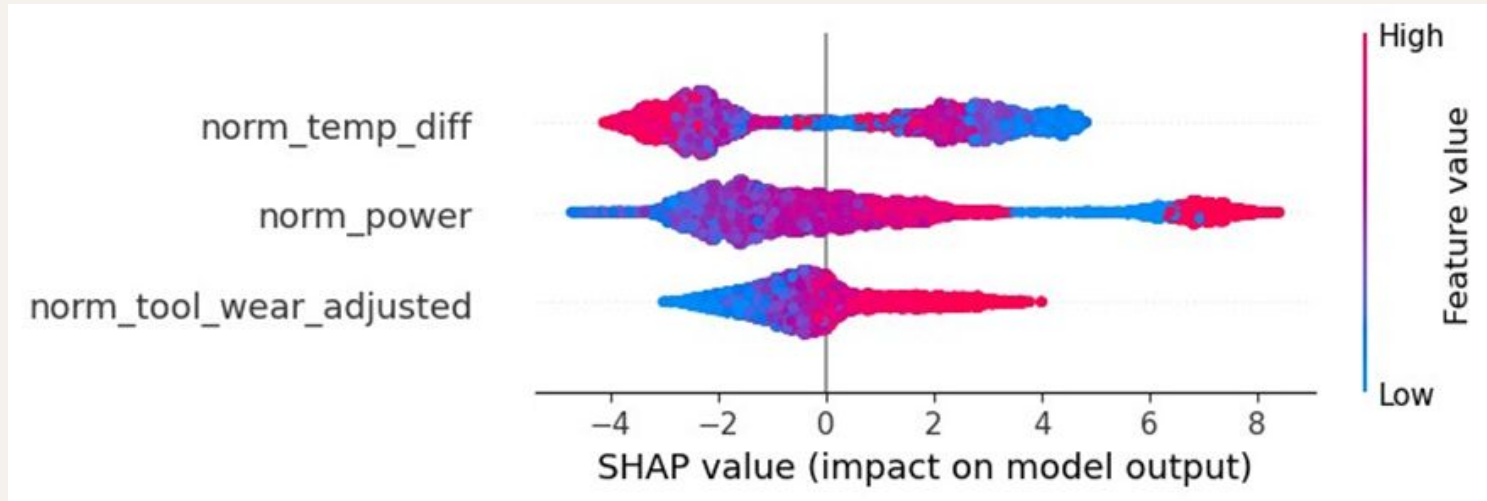
| | |
|---------------------|---------|
| Threshold | 0.20 |
| Accuracy (Training) | .9806 |
| Accuracy (Test) | .9733 |
| Accuracy Difference | .009388 |
| Precision | .9518 |
| Recall | .9927 |
| F1-Score | .9718 |

Final Results - ROC/PRC Curves



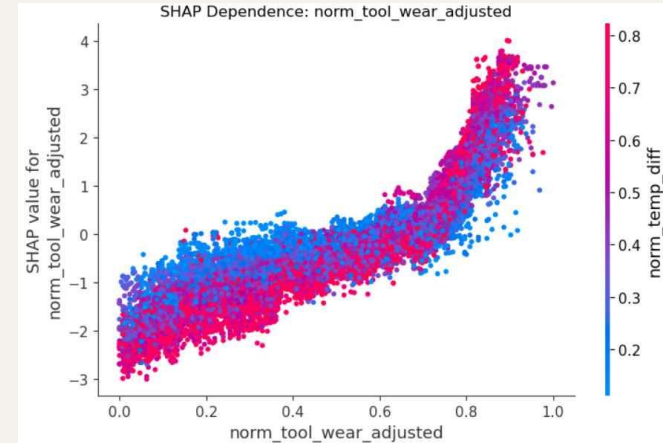
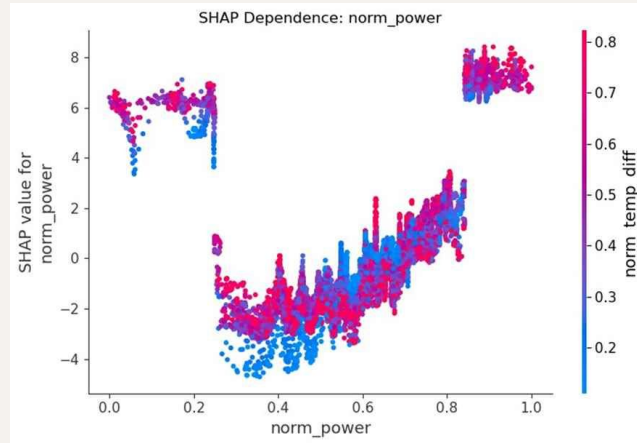
Concerning, but may be an artifact of data being synthetic?

Final Results - SHAP Data



Shows that insufficient cooling (blue on norm_temp_diff), and high power (red on norm_power) contribute to failure

Final Results - SHAP Data



Shows that aside from cooling, power can have a significant influence on failures once out of ideal ranges, and while extreme tool wear can have an effect, current maintenance does a fairly good job of mitigating that risk.

Improvements over first Model Tuning

- Reduced false negatives, still excellent accuracy and small differences between training and test accuracy, indicates not overfitting
- Model exported as JSON to reduce need to rerun expensive training, still need to streamline preprocessing and scaling to use on future data
- Further identified values that contribute the most, may need to explore and understand the SHAP plots more

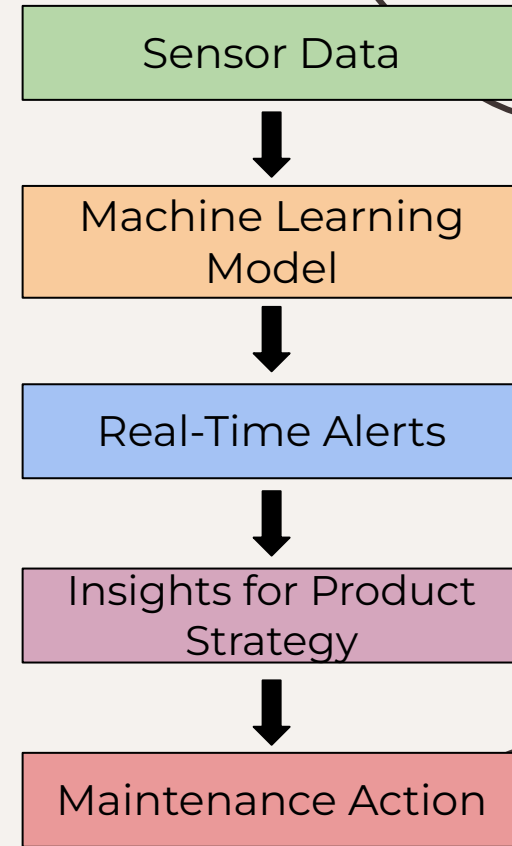
Evaluation

How can we reduce the frequency of factory machine failures by identifying and addressing the most impactful failure causes?

- Target top failure causes (cooling, power issues, tool wear)
- Move from routine to targeted maintenance
- Use real-time monitoring to act before failures
- Invest in continuous machine data tracking
- Use predictions to rank and address failure risks efficiently

Next Steps

- Deploy model in real-time factory systems for proactive failure detection
- Expand model to different machine types and industries
- Use failure insights to guide upgrades for high-risk systems
- Develop product offerings targeting top failure causes:
 - Temperature management solutions
 - Power stabilization tools
 - Tool wear monitoring systems
- Bridge the gap between raw data and actionable maintenance decisions



Citation

*We also used Professor Holly Russo's CDS 303-001 slide deck

[Matzka, Stephan. "Explainable Artificial Intelligence for Predictive Maintenance Applications." 2020 Third International Conference on Artificial Intelligence for Industries \(AI4I\) \(2020\): 69-74.](#)

<https://www.researchgate.net/publication/362517988> REVIEW OF STUDY OF EFFECT OF MISALIGNMENT ON ROTATING SHAFT

<https://www.researchgate.net/publication/340402281> Causes and Impact of Human Error in Maintenance of Mechanical Systems

<https://www.machinemetrics.com/blog/machine-failure>

<https://www.vortec.com/en-us/electronic-equipment-failures-cause-effect-and-resolution?srsId=AfmBOqEMPlapIjrJtRKx0PWhFimiUHiZ5-QihzAYmvhTyKzjPppPjV>

<https://www.graceport.com/blog/top-10-electrical-failures-by-cause>

<https://blog.isa.org/worlds-largest-manufacturers-lose-1-trillion/year-to-machine-failure>

<https://www.rewo.io/the-true-cost-of-downtime-from-human-error-in-manufacturing/>

<https://vectosystem.com/how-much-do-power-quality-disruptions-cost-us-industry/>

<https://eworkorders.com/cmms-industry-articles-eworkorders/dust-threat/>

<https://reliability.thenonstopgroup.com/equipment-malfunction/#:~:text=Unexpected%20equipment%20malfunctions%20can%20result,downtime%20caused%20by%20equipment%20malfunctions.>

<https://www.assemblymag.com/articles/96518-equipment-failure-is-costly-for-manufacturers>

<https://sensemore.io/ensuring-machine-reliability-the-role-of-predictive-maintenance/?srsId=AfmBOoroBs30HAglgmVlf3Lx1mqXzM1YYcbYQBe9DhWhCJ5qr-uFyNRt>

Citation

<https://encord.com/blog/an-introduction-to-balanced-and-imbalanced-datasets-in-machine-learning/>

https://imbalanced-learn.org/stable/references/generated/imblearn.over_sampling.SMOTE.html

<https://www.analyticsvidhya.com/blog/2020/10/overcoming-class-imbalance-using-smote-techniques/>

<https://medium.com/@megha.natarajan/mastering-data-scaling-techniques-visualizations-and-insights-a00b2cb422c2>

<https://ken-hoffman.medium.com/decision-tree-hyperparameters-explained-49158ee1268e>

https://www.geeksforgeeks.org/random-forest-hyperparameter-tuning-in-python/?ref=next_article

<https://www.geeksforgeeks.org/hyperparameters-of-random-forest-classifier/>

<https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/>

https://www.sciencedirect.com/science/article/pii/S2405896320301932?ref=pdf_download&fr=RR-2&rr=92948addac1c974

<https://www.sciencedirect.com/science/article/pii/S2667344424000124>

<https://www.mdpi.com/2075-1702/12/6/357>