# Smart CNC Monitoring: A Data-Driven Modeling Approach to Predict Tool Wear and Clamping Faults

By Imran Aziz

A20550692

## Abstract

Predictive maintenance plays a vital role in smart manufacturing by minimizing unplanned downtime, extending tool life, and improving product quality. This project presents a data-driven modeling approach for CNC machine monitoring, using time-series sensor data collected from 18 milling experiments conducted at the **SMART Lab, University of Michigan**. Each experiment includes high-frequency measurements, such as motor current, velocity, and power, along with metadata describing tool condition, clamping pressure, and the outcomes of visual inspections.

The primary objective is to classify whether the cutting tool used in each experiment is **worn** or **unworn**, based on patterns in the sensor data. Additionally, the project explores the detection of clamping inadequacy by analyzing the relationship between clamping pressure and visual inspection outcomes. Statistical features are extracted from the time-series signals, and appropriate machine learning models are applied to evaluate predictive performance across both tasks.

The results demonstrate the effectiveness of data-driven techniques in identifying key fault conditions in CNC machining. This approach supports the development of intelligent maintenance systems, contributing to more efficient, reliable, and autonomous manufacturing operations.

## Introduction

In modern manufacturing environments, ensuring equipment reliability and product quality is essential for maintaining efficiency and reducing operational costs. Computer Numerical Control (CNC) machines are central to precision manufacturing, yet they are prone to issues such as tool wear and clamping faults, leading to production delays, quality defects, and costly rework. Traditional maintenance strategies—such as reactive or time-based maintenance—often result in unnecessary downtime or premature part replacement.

As part of the transition to **smart manufacturing** and **Industry 4.0**, there is growing interest in data-driven predictive maintenance approaches that leverage sensor data and intelligent algorithms to detect early signs of failure. These methods can help optimize maintenance schedules, minimize unnecessary tool changes, and improve decision-making on the shop floor.

This project aims to develop a predictive monitoring system for CNC machines using time-series sensor data. The primary goal is to classify whether the cutting tool in a given experiment is worn or unworn by identifying patterns in signals such as motor current, velocity, and power. A secondary objective is to investigate clamping inadequacy, using the relationship between clamping pressure and visual inspection outcomes to assess part quality.

The dataset used in this study includes **18 CNC machining experiments** performed on wax blocks, collected at the System-level Manufacturing and Automation Research Testbed

(SMART Lab) at the University of Michigan. Each experiment contains both **metadata (e.g., tool condition, feed rate, clamping pressure)** and high-frequency time-series signals from four motors. Using this rich dataset, the project applies appropriate machine learning models to evaluate predictive performance and uncover actionable insights for smarter CNC operations.

## Dataset Architecture

The dataset consists of both metadata and high-frequency sensor time-series data captured from 18 individual machining experiments. These experiments were performed under varying feed rates, clamping pressures, and tool conditions on **2" x 2" x 1.5"** wax blocks.

**A. Metadata – train.csv**

The train.csv file contains summary information for each of the 18 experiments. It includes:

**Features:**

- feed rate: the cutting speed of the tool relative to the workpiece (mm/s).
- Clamp pressure: the pressure applied to secure the workpiece (bar).

**Labels:**

- Tool condition: the state of the cutting tool (worn or unworn).
- Machining completed: whether the machining was successfully completed without safety interruptions.
- Passed visual inspection: whether the final part passed a visual quality check.

These labels provide the target variables for predictive modeling in tool wear detection and clamping fault analysis.

**B. Sensor Time Series – experiment_01.csv to experiment_18.csv**

Each experiment is recorded in a separate file with high-frequency sensor readings captured at a 100 ms sampling rate. Each file includes over 40 columns representing sensor data from the four motors controlling the X, Y, Z axes, and the spindle. Sensor features include:

- Motor positions (Actual Position and Command Position)
- Velocities, accelerations
- Electrical readings (current feedback, output voltage, output power)
- Control program metadata (M1_CURRENT_FEEDRATE, M1_SEQUENCE_NUMBER)
- Operational stages (Machining Process, e.g., "Layer 1 Up", "Layer 2 Down")

These time-series files are mapped to their corresponding metadata labels using the experiment number (e.g., experiment_01.csv aligns with row 1 in train.csv).

This combination of metadata and raw sensor data enables comprehensive modeling of CNC machine behavior under different tools and clamping conditions.

***Figure 1:*** *S" pattern machined on a 2"×2"×1.5" wax block used in SMART Lab CNC experiments***.**

## Exploratory Data Analysis (EDA) for CNC Sensor and Experiment Metadata

As a preliminary step in the analysis, Exploratory Data Analysis (EDA) was conducted to better understand the characteristics and structure of the dataset. The dataset consisted of time-series sensor readings obtained from eighteen CNC machining experiments, each performed on a 2" × 2" × 1.5" wax block and recorded in individual Excel files. These files were merged with a metadata file (train.csv) containing key experimental attributes such as feedrate, clamping pressure, tool condition (worn or unworn), and the outcome of the visual inspection.

The code was designed to iterate through all experiment files, read each into memory, and append corresponding metadata labels by matching experiment numbers. Basic statistical summaries and distributions were then plotted using matplotlib and seaborn to examine the balance of class labels and continuous variables such as feedrate and clamp pressure. For example, count plots were used to visualize the distribution of worn vs. unworn tools and parts that passed vs. failed visual inspection.

Furthermore, to understand interdependencies between recorded sensor channels, a correlation matrix was computed using only numeric columns from the aggregated dataset. The resulting heatmap highlighted relationships among signals such as motor current, power output, and axis velocities. To explore the time-based behavior of individual sensors, sensor-specific line plots were generated for select experiments, contrasting worn and unworn tool conditions. These visualizations included output current trends across the X, Y, Z axes and the spindle motor.

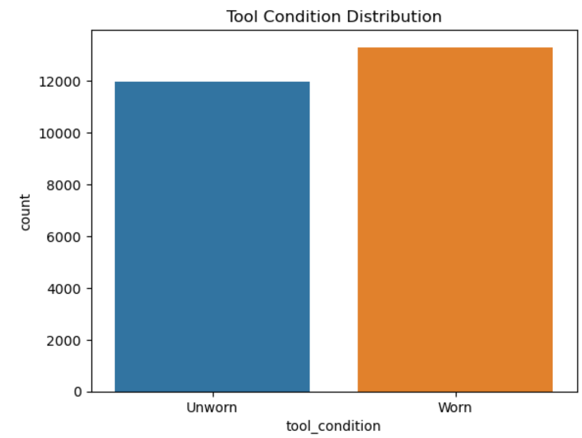


***Figure 2:*** *Distribution of Tool Condition Labels*

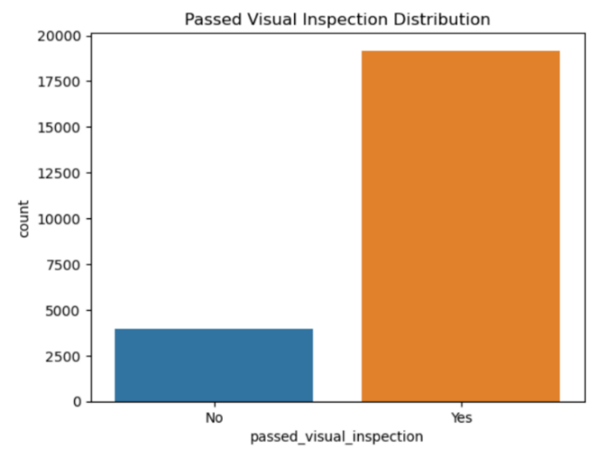


***Figure 3:*** *Distribution of Visual Inspection Outcomes*

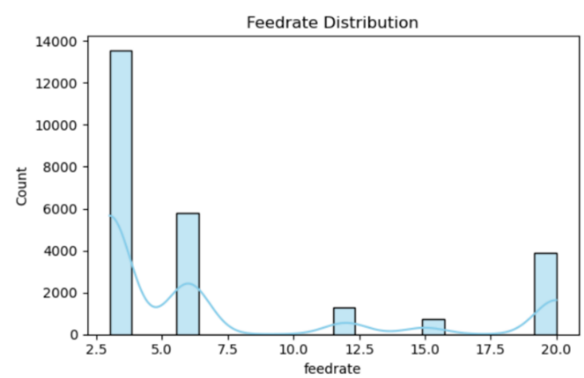


***Figure 4:*** *Feedrate Distribution Across All Experiments*

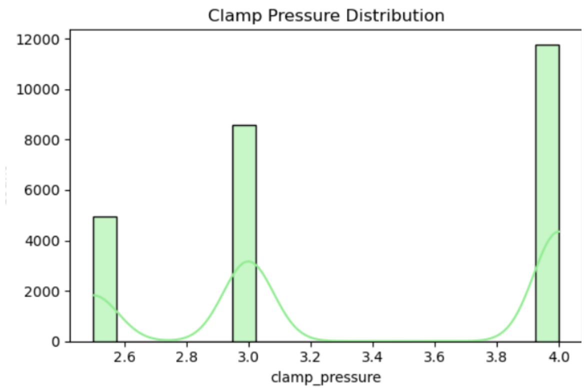


***Figure 5**: Clamp Pressure Distribution Across All Experiments*

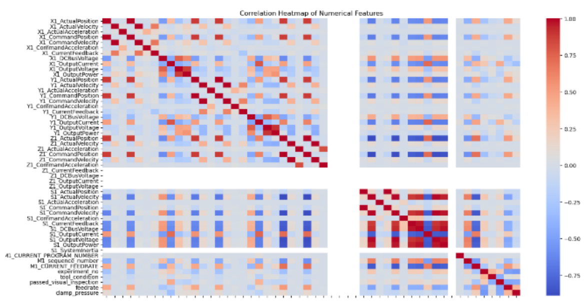

***Figure 6:** Correlation Heatmap of Numerical Features*

The exploratory data analysis begins with **Figure 2**, which presents the distribution of tool condition labels across the dataset. The chart shows a nearly balanced number of samples labeled as “Worn” and “Unworn,” which is beneficial for supervised classification tasks. A balanced dataset like this reduces the risk of bias during training and allows the machine learning model to learn the characteristics of both classes effectively.

In **Figure 3**, the distribution of parts that passed or failed visual inspection is illustrated. Unlike the tool condition distribution, this outcome shows a noticeable class imbalance, with a significant majority of parts passing inspection. This imbalance could affect the model’s sensitivity to the minority class if this output is used as a target variable. Techniques such as resampling or weighted loss functions may be considered when modeling this outcome.

**Figure 4** provides insight into the distribution of feedrate values. The histogram combined with a KDE curve reveals that feedrate values were applied in distinct steps or modes across experiments, likely reflecting controlled machining parameters. The presence of strong peaks at certain values suggests that experimental setups were systematically varied, which could help analyze how changes in feed rate influence tool wear and final product quality.

Similarly, **Figure 5** focuses on clamp pressure, another important input parameter. The distribution shows three dominant clamp pressure settings—2.5, 3.0, and 4.0 bar—each represented by distinct peaks. These fixed values confirm that clamp pressure was also a controlled variable during experimentation. Understanding the frequency and effect of each pressure level is essential for determining its influence on machining success and tool degradation.

Lastly, **Figure 6** presents a correlation heatmap of all numerical features. The map highlights patterns of linear association among variables, with dense red diagonal lines indicating perfect correlation of features with themselves, and various off-diagonal blocks suggesting interrelated sensor measurements. For example, actual and command values of position, current, or voltage tend to be highly correlated, reflecting how motor behavior is tightly regulated during CNC operations. This visualization is useful not only for understanding feature relationships but also for identifying redundant variables that can be excluded during feature selection to improve model efficiency and reduce overfitting.

Together, these figures provide a comprehensive view of the data, validating the quality and consistency of the experimental setup while uncovering key trends and relationships that inform downstream modeling.

## Sensor Signal Trends Visualization

To better understand the dynamic behavior of the CNC machine under varying tool conditions and process parameters, time-series visualizations of key sensor signals were generated across all experiments. These plots illustrate how measurements such as motor output current, spindle power, and axis velocity evolve during machining. By overlaying signals from multiple experiments, it becomes possible to visually compare patterns associated with worn and unworn tools. These comparisons are critical for identifying temporal signatures of tool degradation, detecting process anomalies, and validating the consistency of experimental settings. The insights derived from these trends not only inform feature engineering for machine learning but also offer practical diagnostic value in real-world manufacturing scenarios.


***Figure 7:*** *Legend for Experiments*

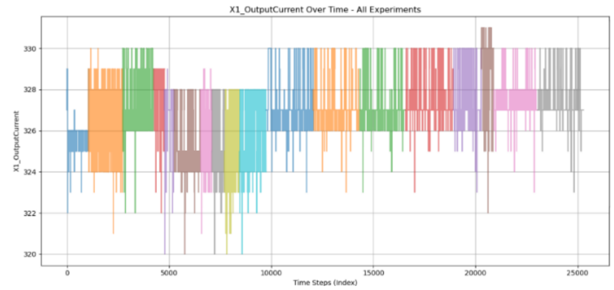


***Figure 8****: Overlayed time-series plot of X1 output current across all 18 experiments, categorized by tool condition (Worn vs. Unworn).*

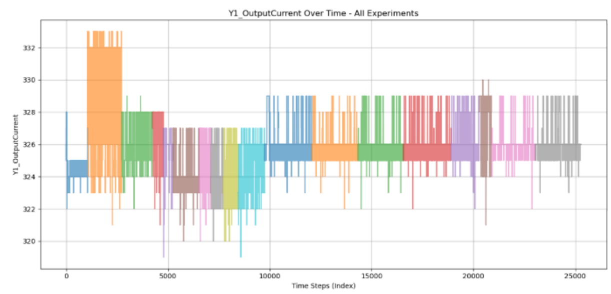


***Figure 9:*** *Time-series plot of Y1 output current across all experiments.*

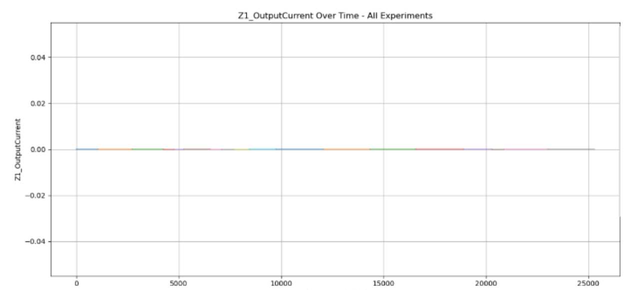


***Figure 10:*** *Time-series plot of Z1 output current across all experiments.*

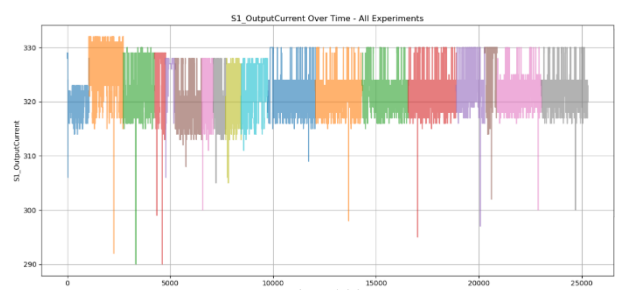


***Figure 11:*** *S1 Output Current across all experiments plotted over time.*

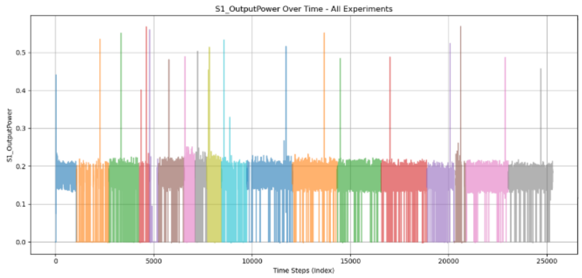


***Figure 12:*** *S1 Output Power across all experiments over time.*

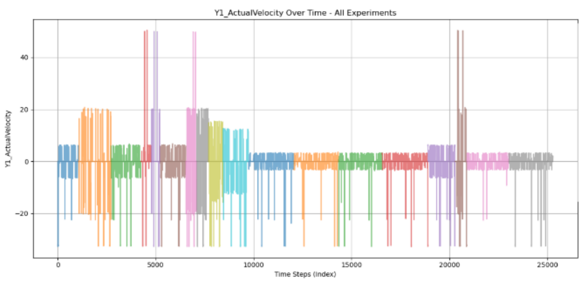


***Figure 13:*** *Y1 Axis Actual Velocity over Time across all experiments.*

## Sensor Trends for Spindle and Axis Monitoring

**Figure 8** presents the time-series variation of the X-axis motor's output current across all 18 experiments, with each line representing an individual experiment. The fluctuations in current values correspond to the changing mechanical loads experienced by the X-axis during machining. Higher current spikes may indicate periods where the tool encountered greater material resistance, tougher sections, or deeper cuts, while flatter segments suggest periods of lighter engagement or repositioning. Observing these variations is critical for diagnosing whether tool wear or improper feed rates contributed to increased machine load along the X-axis.

The Y-axis output current trends shown in **Figure 9** similarly reveal how motor load changes throughout the machining process. Movement along the Y-axis is often responsible for lateral tool engagement during carving operations, and variations in current levels can reflect shifts in material contact, feedrate changes, or adjustments in cutting direction. Pronounced peaks in Y1 current could be attributed to tougher material sections or slight misalignments, offering early warnings about tool degradation or fixture stability. Consistent load patterns, on the other hand, suggest stable and repeatable machining behavior.

**Figure 10** tracks the Z-axis motor's output current, which remains nearly flat and close to zero throughout all experiments. The Z-axis typically controls vertical movements such as plunging or retracting the tool. The near-constant low current indicates that vertical motion was minimal or highly controlled during these experiments. This suggests that most of the machining involved horizontal layer carving with little to no vertical drilling or aggressive vertical engagement, confirming the planar nature of the experimental design focused on layered tool paths.

The spindle motor's current output over time is shown in **Figure 11**, offering crucial insights into the tool's active cutting behavior. Since the spindle directly drives the cutting tool, its current load reflects the cutting forces experienced during machining. Stable current during cutting implies consistent tool-material interaction, whereas abrupt increases could indicate contact with harder sections, tool dulling, or chip accumulation. Tracking spindle current is vital for early detection of abnormal machining conditions, tool degradation, and predicting imminent failure events.

**Figure 12** illustrates the power consumption trends of the spindle motor across all experiments. Spindle power represents the electrical energy being delivered to maintain rotational speed under varying cutting loads. Peaks in the output power curve could correspond to intensive cutting phases, sudden increases in material density, or inefficient cutting when the tool becomes worn. A gradual rise in spindle power across machining cycles could serve as an early indicator of tool wear progression, even before visible surface defects appear on the machined part.

The Y1 actual velocity plot, as shown in **Figure 13,** represents the real-time movement speed of the Y-axis actuator throughout the machining process. It captures the magnitude and direction of motion, with changes reflecting acceleration, deceleration, and reversals during tool path execution. Fluctuations in velocity curves could be associated with transitions between different machining layers, tool repositioning, or adjustments to avoid collisions. Analyzing velocity trends is critical for ensuring smooth, stable operations, as sudden drops or erratic velocities can induce mechanical stresses that accelerate tool wear and degrade machining precision.

## Sensor Signal Feature Extraction and Initial Classification

After loading and preparing the data from all 18 experiments, feature engineering was performed to extract meaningful information from the raw sensor signals. For each numeric sensor reading, statistical features such as mean, standard deviation, root mean square (RMS), maximum, and minimum were computed. This transformation condensed large time-series datasets into compact representations that captured essential patterns relevant to machine behavior. To further refine the dataset, feature selection was applied using Random Forest's feature importance scores, which helped identify and retain the most influential features while discarding less relevant or noisy inputs. This ensured that the models would focus on the strongest signals in the data, improving both training efficiency and model interpretability.

Following feature engineering and selection, **Logistic Regression** was chosen as the baseline model for the classification tasks. Due to its simplicity, interpretability, and strong performance on structured datasets, Logistic Regression was used to build two predictive models: one to classify tool wear conditions (worn vs. unworn) and another to detect clamping faults based on visual inspection outcomes. The models were trained using a 70/30 train-test split and evaluated with standard classification metrics, including accuracy, precision, recall, and F1-score. This baseline model establishes a clear benchmark for performance, providing a foundation for future work with more complex machine learning models.

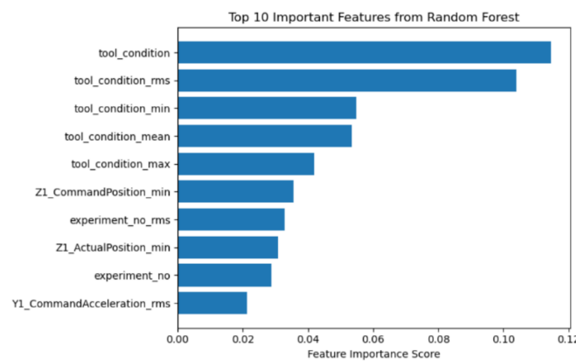


***Figure 14:*** *Top 10 important features ranked by Random Forest feature importance scores.*

Logistic Regression Results: Tool Wear Prediction

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Class 0 | 0.400000 | 0.670000 | 0.500000 | 3 |
| Class 1 | 0.000000 | 0.000000 | 0.000000 | 3 |
| Macro Avg | 0.200000 | 0.330000 | 0.250000 | 6 |
| Weighted Avg | 0.200000 | 0.330000 | 0.250000 | 6 |

***Figure 15:*** *Logistic Regression - Tool Wear Prediction*

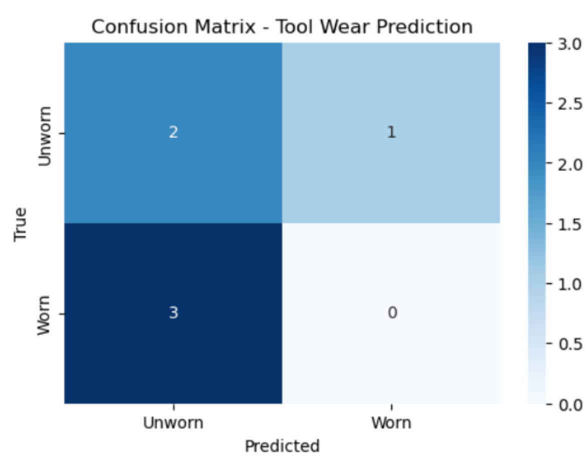


***Figure 16:*** *Confusion matrix illustrating the performance of Logistic Regression for tool wear classification.*

**Figure 14** presents the top 10 most important features identified by the Random Forest model during the feature selection process. It is evident that attributes related to tool condition, including its raw value, root mean square (rms), minimum, mean, and maximum statistics, are highly influential in predicting tool wear. Additional features such as the minimum value of Z1 axis command position and experiment number statistics, also show moderate importance. This feature ranking is crucial because it helps narrow down the variables most associated with the tool's performance and health status, guiding the model to focus on the most predictive signals.

**Figure 15** shows the detailed classification metrics for the Logistic Regression baseline model used to predict tool wear. The results indicate that while the model achieved a moderate precision and recall for the 'Unworn' class, it failed to predict the 'Worn' class entirely, reflected by zero precision, recall, and f1-score for that class. The overall accuracy stands at 33%, which is relatively low, confirming that the model struggles with class imbalance and the complexity of the problem when using a simple baseline without optimization.

**Figure 16** depicts the confusion matrix for tool wear prediction, where the model correctly predicted two instances of 'Unworn' tools but misclassified three instances of 'Worn' tools as 'Unworn'. Only one 'Unworn' tool was wrongly classified as 'Worn'. This matrix further highlights the model's difficulty in detecting worn tools, reinforcing the need for more advanced models or better feature engineering and balancing strategies to improve the predictive performance.

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| Class 0 | 0.5 | 1 | 0.67 | 1 |
| Class 1 | 1 | 0.75 | 0.86 | 4 |
| Macro Avg | 0.75 | 0.88 | 0.76 | 5 |
| Weighted Avg | 0.9 | 0.8 | 0.82 | 5 |

***Figure 17**: Logistic Regression - Clamping Fault*

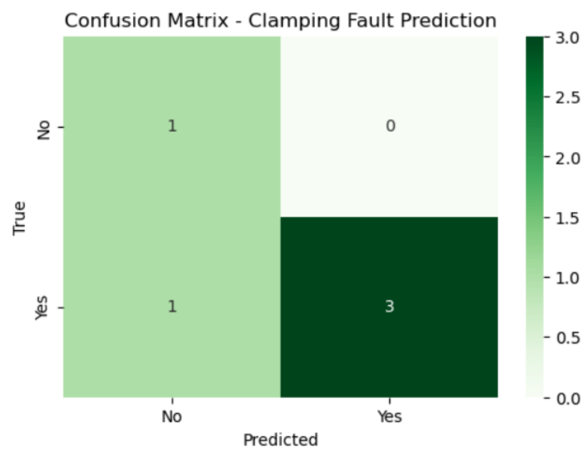


***Figure 18:** Confusion Matrix - Clamping Fault Prediction*

The table shown in **Figure 17** summarizes the logistic regression model's performance in predicting clamping faults. Class 0 corresponds to no clamping fault, and Class 1 indicates the presence of a clamping fault. The model achieves high precision (1.0) and a strong F1-score (0.86) for Class 1, demonstrating its ability to identify clamping faults when they occur correctly. The overall accuracy is **80%**, and both the macro and weighted averages reflect balanced performance, with macro F1-score at **0.76** and weighted F1-score at **0.82**. This highlights the model's reliability in predicting clamping conditions despite a small dataset.

The confusion matrix in **Figure 18** provides a visual representation of the classification results for clamping fault detection. It shows that out of the five samples, the model correctly predicts three cases with clamping faults (True Positives) and one case without a fault (True Negative). However, there is one False Negative, where a fault was present but not detected. Overall, the model captures the majority of true cases, aligning well with the performance metrics summarized in the table.

## Experimenting with Superior Machine Learning Models

After establishing Logistic Regression as a baseline model, **XGBoost** was introduced to further enhance the predictive performance. XGBoost, known for its robustness and ability to handle complex, non-linear relationships, was particularly suited for this dataset, where multiple sensor-derived features interact. While Logistic Regression provided a quick and simple benchmark, it is limited by its linear assumptions. In contrast, XGBoost applies gradient boosting on decision trees, allowing it to capture more intricate patterns in the data. By comparing the performance of XGBoost with Logistic Regression, it becomes evident whether more sophisticated machine learning models are necessary to achieve higher accuracy and better fault detection reliability.

==== XGBoost: Tool Wear Prediction ====

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0 | 0.00 | 0.00 | 0.00 | 2 |
| 1 | 0.60 | 1.00 | 0.75 | 3 |
| accuracy | | | 0.60 | 5 |
| macro avg | 0.30 | 0.50 | 0.38 | 5 |
| weighted avg | 0.36 | 0.60 | 0.45 | 5 |

***Figure 19:*** *XGBoost - Tool Wear Prediction*

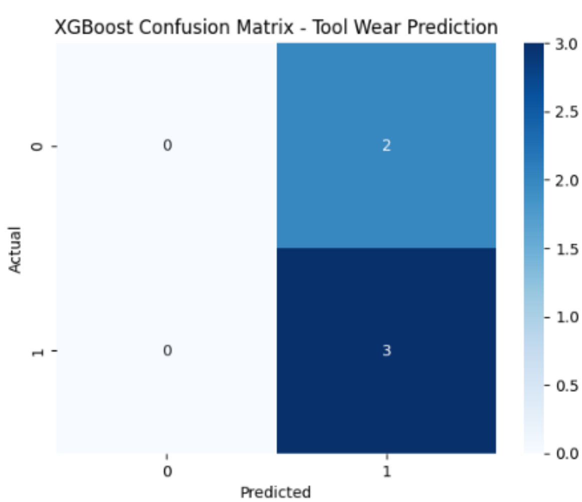


***Figure 20:*** *XGBoost Confusion Matrix - Tool Wear Prediction*

==== XGBoost: Clamping Fault Prediction ====

| | precision | recall | f1-score | support |
|---|---|---|---|---|
| 0.0 | 0.80 | 1.00 | 0.89 | 4 |
| 1.0 | 0.00 | 0.00 | 0.00 | 1 |
| accuracy | | | 0.80 | 5 |
| macro avg | 0.40 | 0.50 | 0.44 | 5 |
| weighted avg | 0.64 | 0.80 | 0.71 | 5 |

***Figure 21****: XGBoost - Clamping Fault Prediction*

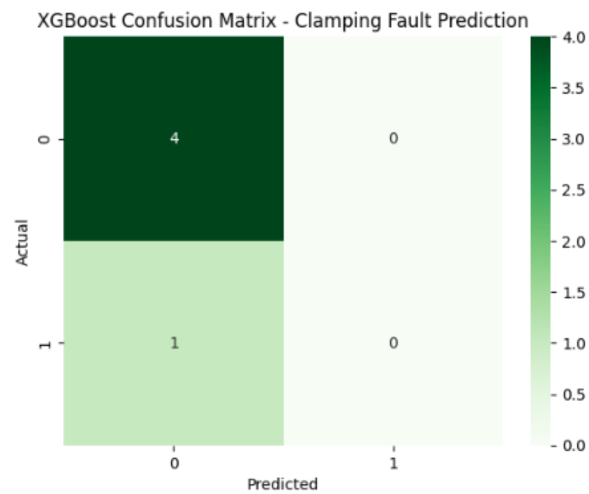


***Figure 22:*** *XGBoost Confusion Matrix - Clamping Fault Prediction*

To further improve the model's performance and ensure its robustness, hyperparameter tuning and cross-validation were applied to the XGBoost classifier. Grid Search with 5-Fold Cross-Validation was used to systematically explore combinations of important hyperparameters such as the number of trees (n_estimators), tree depth (max_depth), learning rate, and subsampling rate. Cross-validation was necessary to avoid overfitting to a single train-test split and to ensure that the model generalizes well across different subsets of the data. Hyperparameter tuning helped identify the best set of model settings that could maximize the predictive performance, leading to a more accurate and reliable classification of tool wear and clamping faults across all experiments.

## Model Comparison

In this study, we compared the performance of **Logistic Regression** and **XGBoost** models for two classification tasks: **Tool Wear Prediction** and **Clamping Fault Detection**. For Tool Wear Prediction, XGBoost clearly outperformed Logistic Regression across all metrics. The accuracy improved from **33%** with Logistic Regression to **60%** with XGBoost, and the F1-score improved from **0.25** to **0.45**, demonstrating that XGBoost was better at balancing precision and recall. Logistic Regression struggled particularly with Class 1 (worn tools), failing to predict them correctly, whereas XGBoost showed significantly better recall for the worn class. These results confirm that a more sophisticated, non-linear model like XGBoost is better suited for capturing complex patterns related to tool wear behavior compared to a simple linear model like Logistic Regression.

For Clamping Fault Detection, however, the performance comparison was more balanced. Both Logistic Regression and XGBoost achieved a similar **accuracy of 80%**, but Logistic Regression exhibited slightly higher **precision** and **F1-score**, indicating it was more reliable in correctly predicting clamping faults without excessive false positives. This suggests that for simpler binary classification tasks where the data is relatively well-separated, Logistic Regression can perform competitively against more complex models. Overall, while XGBoost offered a clear advantage in predicting tool wear, Logistic Regression remained a strong baseline for clamping fault detection, highlighting the importance of selecting models based on the specific nature and complexity of the prediction task.

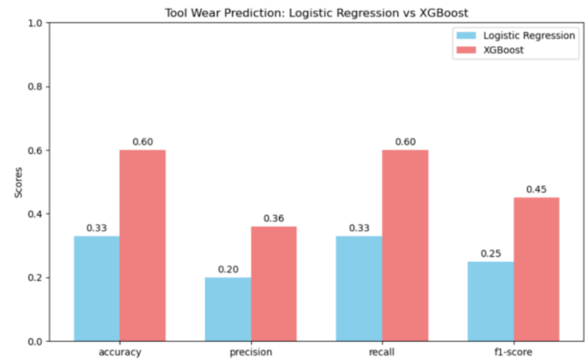


***Figure 23:*** *Tool Wear Comparison*

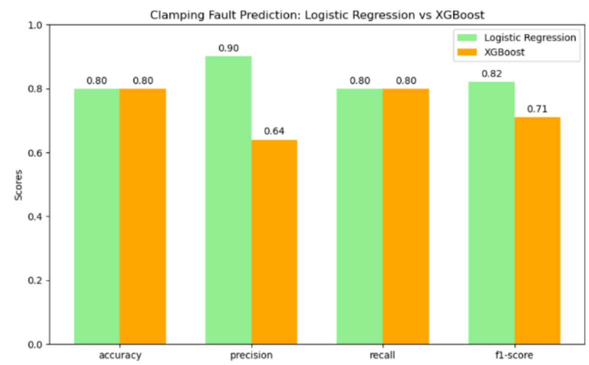


***Figure 24:*** *Clamping Fault Comparison*

## Error Analysis

During the evaluation of model performance, it was observed that the classifiers, particularly logistic regression and even XGBoost to some extent, struggled with distinguishing certain cases, for example, misclassifying worn tools as unworn. This confusion likely stems from overlapping sensor patterns, limited data availability, or inherent noise in the sensor readings, which reduces the clarity between the two classes. In some instances, minor fluctuations in current, voltage, or acceleration features were not distinct enough to allow the model to separate classes confidently. Recognizing these challenges highlights the need for either better feature engineering, more robust sensors, or an expanded dataset in future work.

## Final Reflection and Future Work

This project successfully demonstrated that machine learning models like logistic regression and XGBoost can be used to predict tool wear and clamping faults based on sensor data. However, to further improve the robustness and accuracy of these models, future work should include conducting more experiments and introducing a greater variety of sensor types to capture additional aspects of machine behavior. Moreover, if the dataset grows significantly in size, exploring deep learning models such as LSTM networks or CNNs could be a promising direction. These architectures can capture temporal and complex feature patterns more effectively, leading to improved predictive performance. Overall, the project sets a solid foundation, with clear avenues for enhancement.

## References

[1] M. Soori, B. Arezoo, and R. Dastres, "Machine learning and artificial intelligence in CNC machine tools: A review," University of Kyrenia, Amirkabir University of Technology, Near East University.

[2] Y. Guo, Y. Sun, and K. Wu, "Research and development of monitoring system and data acquisition of CNC machine tool in intelligent manufacturing.

[3] M. Holub, O. Andrs, V. Stepanek, J. Kroupa, R. Huzlik, and J. Tuma, "Experimental study of operational data collection from CNC machine tools for advanced analysis.

[4] O. F. Fapetu, O. R. Bodede, O. R. Ayodele, O. C. Oloye, M. D. Olupona, and A. O. Durotoluwa, "A quantitative analysis: Assessment of data-driven decisions in CNC machining.