Reflection on Project: Tool Wear Prediction for CNC Machining

This project focused on analysing and predicting **tool wear** during CNC machining using experimental data collected from the University of Michigan SMART Lab. As a mechanical engineer, my motivation was to link data-driven methods with the physical understanding of machining processes, which are central to my PhD research on the optimisation of hybrid additive manufacturing and CNC machining.

Before starting, I had no prior experience with Python or data science libraries. My first challenge was setting up the environment, understanding the project structure from the Bristol example repository, and learning how to run and test code locally. Working through these steps gave me a practical understanding of reproducibility and version control with GitHub.

The **data preparation** required combining several experiment CSV files, cleaning missing or constant columns, and normalising variables. I learned to use pandas for handling data frames and pathlib for working with file paths. Reducing the dataset from 18 to 5 experiments made it easier to test code, although it limited the representativeness of the model. This trade-off between complexity and clarity helped me understand why smaller, well-structured datasets are useful for learning.

For **exploratory data analysis (EDA)**, I created correlation heatmaps and visualised distributions of feed rate, clamp pressure, and current. These plots showed how the machine parameters influence wear and confirmed trends described in related studies [3,4]. Implementing visualisation in matplotlib and seaborn taught me how to interpret patterns and relationships numerically and graphically.

I later expanded the analysis with **PCA** and **K-means clustering**, which helped me visualise how sensor data could group into distinct operating conditions. These methods provided insight into data structure and variability. Finally, using a **LightGBM model** for classification linked the exploratory stage to predictive analytics, demonstrating how features such as current, feed rate, and clamp pressure relate to the tool's wear state.

The main difficulty was translating engineering intuition into data operations. Unlike traditional mechanical experiments, machine learning requires preprocessing, model validation, and evaluation. By following a modular notebook structure, I learned to document each step clearly, making the analysis reproducible for others.

Through this project, I gained three key skills:

1. **Practical Python use** for data cleaning, visualisation, and modelling.
2. **Understanding of predictive maintenance** concepts through applied machine learning.

3. **Confidence using GitHub and Jupyter notebooks** to structure a scientific workflow.

Future improvements would include restoring the full dataset, adding time-series analysis to capture tool wear progression, and comparing multiple algorithms. Despite starting with no programming background, I now feel confident in extending this foundation to more advanced data-driven studies within my mechanical engineering research.

Sources

[1] Kaggle, *Tool Wear Detection in CNC Milling*.
[2] Wu, D. et al., “Cloud-based machine learning for predictive analytics: Tool wear prediction in milling,” *IEEE Big Data*, 2016.
[3] Chiu, S.-M. et al., “Development of Lightweight RBF-DRNN and Automated Framework for CNC Tool-Wear Prediction,” *IEEE Trans. Instrum. Meas.*, 2022.
[4] Sun, J. et al., “Tool digital twin based on knowledge embedding for precision CNC machine tools,” *Journal of Manufacturing Systems*, 2025.