A study on predictive maintenance using edge intelligence

Master's thesis in Production Engineering

PRAVEEN PACHA
SAI BHARATH KOLISETTY
DEPARTMENT OF INDUSTRIAL AND MATERIALS SCIENCE
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden 2022
www.chalmers.se

Master's thesis 2022

A study on predictive maintenance using edge intelligence

PRAVEEN PACHA

SAI BHARATH KOLISETTY

CHALMERS
UNIVERSITY OF TECHNOLOGY

Department of Industrial and Materials Science
Chalmers University of Technology
Gothenburg, Sweden 2022

A study on predictive maintenance using edge intelligence
PRAVEEN PACHA & SAI BHARATH KOLISETTY

Academic Supervisors: Ebru Turanoglu Bekar and Martin Dahl, Chalmers University of Technology
Examiner: Anders Skoogh, Chalmers University of Technology

Master's Thesis 2022
Department of Industrial and Materials Science
Chalmers University of Technology
SE-412 96 Gothenburg
Telephone +46 31 772 1000

Typeset in LaTeX
Printed by Chalmers Reproservice
Gothenburg, Sweden 2022

A study on predictive maintenance using edge intelligence
PRAVEEN PACHA & SAI BHARATH KOLISETTY
Department of Industrial and Materials Science
Chalmers University of Technology

Abstract

Production systems have been following a reactive type of approach for maintenance activities, but in recent times, the interest has shifted towards the development of more proactive approaches to avoid failures. This has resulted in increased industrial data collection through the deployment of information and communication technologies. With this advent, there is also an increase in Predictive Maintenance (PdM) solutions. Developing Machine Learning (ML) models to perform PdM activities have become a recent challenge and a popular research area under Industry 4.0. Therefore, this thesis takes its roots from this emerging area of data-driven decision making for PdM with the support of key enabling technologies of Industry 4.0.

This thesis aims to analyze high-dimensional data coming from Edge devices and investigate what type of predictive algorithms can be designed to implement PdM at the industrial level. Data collected by the Edge device from two Manufacturing companies were analyzed and different ML models were developed for both cases. Further, a comparative study between the cases was presented to show the importance of faulty data to develop a better ML model for PdM. Finally, some recommendations were provided for the successful implementation of the ML model in both companies by using the advantages of the Edge device. This provides a concrete platform for future research in the area of handling missing data and implementation of Edge Intelligence in a manufacturing setup.

Keywords: Edge intelligence, Faulty data, Machine learning, Manufacturing, Predictive Maintenance, Smart Maintenance

Acknowledgements

We take this occasion to offer our sincere gratitude to everyone who assisted us during our master's thesis research. We would like to thank our academic supervisor, Dr. Ebru Turanoglu Bekar whose insights and guidance was very helpful in providing the directions in this master thesis tenure. We would also like to thank Dr. Martin Dahl and Assoc Prof Alexander Karlsson for their valuable suggestions for this thesis.

We would like to extend our gratitude to Johan Balkhammar, Jorgen Kastebo, Robert Andersson Jarl and Urban Broström for providing us with the necessary information and giving us insights from the manufacturing companies. We would also like to thank Bernd Poetzsch, Carl Von Rosen, Thomas Sundqvist and Muhammad Ahmer from the manufacturing companies for their thoughts and insights into this thesis.Thanks also to the Produktion 2030 Strategic Innovation Program funded by VINNOVA for their funding of the research project PACA-Predictive Maintenance using Advanced Cluster Analysis (Grant No. 201900789), which this thesis has been conducted.

Our grateful thanks to our examiner Prof. Anders Skoogh for his constructive feedback and suggestions for this thesis. Finally, we would like to thank all of our peers and family for their good wishes in helping us to complete this thesis.

Praveen Pacha, Gothenburg, June 2022
Sai Bharath Kolisetty, Gothenburg, June 2022

List of Acronyms

Below is the list of acronyms that have been used throughout this thesis listed in alphabetical order:

AI	Artificial Intelligence
CRISP-DM	Cross-Industry Standard Process for Data Mining
DT	Decision Tree
EC	Edge Computing
EDA	Exploratory Data Analysis
FN	False Negative
FP	False Positive
GNB	Gaussian Naive Bayes
IoT	Internet of Things
IIoT	Industrial Internet of Things
ML	Machine Learning
PdM	Predictive Maintenance
TN	True Negative
TP	True Positive
RF	Random Forest

Contents

List of Figures

List of Tables

1
Introduction

Production systems have been following a reactive type of approach for maintenance activities, but in the recent times, the interest has shifted towards the development of more proactive approaches to avoid failures (Lundgren et al., 2021).The Fourth industrial revolution also called Industry 4.0 is the phase where the usage of automation of the third industrial revolution is being enhanced by incorporating autonomous and smart systems where data and advanced data analytics (i.e., big data, artificial intelligence(AI)/ML) can be used (Marr, 2018). This has resulted in increased industrial data collection through the deployment of information and communication technologies, specifically intelligent devices (i.e., Industrial Internet of Things(IIoT) sensors, edge devices and computing). With this advent, there was also an increase in PdM solutions (i.e., machine health (condition) monitoring, anomaly detection, and remaining useful life estimation) by using this industrial data. PdM is defined as a set of techniques designed to help determine the condition of the equipment and estimate future failures. These estimations are further used to schedule the maintenance activities by means of smart scheduling of maintenance actions which helps in avoiding unexpected failures or at least mitigating the effects of it. PdM involves analyzing the data from the machines and extracting meaningful insights by using analytical tools. Further, develop a ML model to use this data and predict future failures. The data collection is done by the use of sensors that have low capabilities of data processing. This has led to the development of Edge Intelligence where the Edge device acts as a sensor and has many more functionalities like data processing, data cleaning, etc. Further, it can also be used to deploy the ML model for local decision support. The use of Edge Intelligence is of key importance in this thesis as it will help in providing real-time decision support. Analyzing and developing these models by incorporating them with Edge Intelligence is a recent challenge for companies and a popular area of research under Industry 4.0.

The project PACA (Predictive Maintenance using Advanced Cluster Analysis) is an ongoing research project in the Department of Industrial and Materials Science at the Chalmers University of Technology under Produktion 2030. This project aims to develop predictive algorithms to increase precision and make the data understandable for the purpose of effective maintenance planning. Therefore, data from real-world cases in the industry are collected from several data sources and machines to identify interesting patterns and to develop an understanding of how the patterns can be used to predict future conditions of machines by using AI/ML. The industrial partners of this project want to analyze the data obtained from the Siemens

Sinumerik Edge device for implementing machine condition monitoring as the core application of PdM. Therefore, this thesis has been produced from the ongoing research by taking a note of the business goals.

Research on the theoretical background is done to understand the recent progress in the field of Edge Intelligence and ML which is presented in chapter 2. Chapter 3 gives an overview of the methodology adopted to accomplish the goals of this thesis. The results of the thesis for both the companies are presented in chapter 4. Further discussion and implementation strategies for the companies are presented in chapter 5. Chapter 6 concludes the outcomes of this thesis.

1.1 Aim and Research Questions

This thesis aims to analyze high-dimensional data coming from Edge devices and investigate what type of predictive algorithms can be designed to implement PdM at the industrial level. Further, perform a comparative study between two cases to understand the importance of faulty data to develop a better ML model for predictive analysis. Knowing the aim of this thesis and the industrial requirements, the following research questions are specified for investigation in this thesis:

- *RQ1: How can the data collected from the edge device be used to analyze the current condition of the machine and further develop an ML model for performing PdM?*
- *RQ2: What will be the impact of faulty data in developing an ML model for achieving accurate predictions?*

1.2 Objectives

Knowing the aim of this thesis, the following objectives are identified:

- Understand the requirements of the stakeholders involved.
- Describe data and prepare a data quality report.
- Formulate potential PdM decision-making solutions which can be implemented in the company based on the given data.
- Define the desired output from the models as per the case requirements.
- Apply pre-processing and feature engineering methods.
- Describe the type of analysis (supervised or unsupervised) and select the algorithm(s) in line with the type of analysis to apply.
- Design the selected algorithm(s) and demonstrate the effects of the developed algorithm(s).
- Perform a comparative study between the cases and provide insights about the use of faulty data in data preparation and model development phases.

1.3 Problem Description

This section will give the reader an overview of the machine setup used in Manufacturing company A and Manufacturing company B. Further, the data acquisition system will be explained along with the problem to be addressed in both cases. The names of the companies involved are not disclosed due to confidentiality purposes.

1.3.1 Manufacturing Company A

Manufacturing Company A works with the manufacturing and assembly of the engine for cars. CNC machines are used to manufacture the cylinder blocks, camshafts, pistons and other components in the engine. In this thesis, we would be dealing with one of the CNC machines which are used to manufacture cylinder blocks for different variants of the engine. The CNC machine is a single channel 5-axis machine provided by GROB Group. The axis involved are X, Y, Z, A and B, where X, Y and Z are the linear axes, A is the linear and rotational axis and B is the rotational axis. Figure 1.1 shows a schematic of the axis position in the CNC machine. This machine was installed around three years back, which means it is a new machine and is very healthy. The production on this machine usually runs for 24 hours but in the recent days, it is not used up to its capacity due to decrease in production.

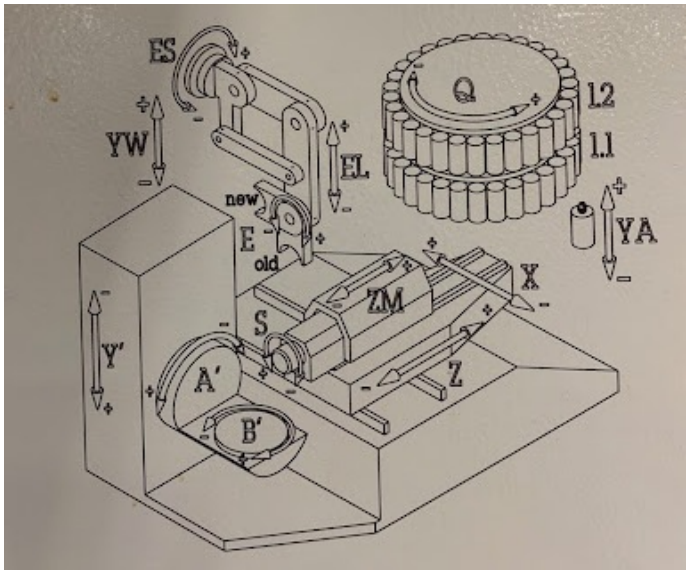

Figure 1.1: Schematic of CNC machine

Siemens Edge device is installed on this CNC machine to extract the axis movement data. The data collection process is not performed during the manufacturing of parts rather it is done once in a weekwhen the production is paused. Application of Siemens Edge device in this case provides the users with seven different tests to monitor the condition of the machine. Among these seven tests, Manufacturing Company A has selected five tests (Equability, Friction, Backlash, Stiffness and Signature) to monitor their CNC machine. The description of these tests is provided in section 2.4.2. These tests have their own specific setup of torque and movement

values for every axis. The Edge device is programmed to collect the data for every test of a specific axis and provide it in the form of a JSON file.

It is essential to calibrate the axis before any part is manufactured to avoid quality issues. Initially, the axis calibration check was carried out manually. This has its own drawbacks like miscalibration is detected only if the variation is too large. Currently, the Siemens application is used to process the data from the Edge device and provide decision support. This process also consumes time as it is not done in real-time. The problem addressed, in this case, is to analyze the raw test data provided by the Edge device and provide information on the calibration of the axis. Further, automate the process of data collection and provide decision-making support in real-time by developing a ML model suitable for this case.

1.3.2 Manufacturing Company B

Manufacturing Company B works with the manufacturing and assembly of bearings for all types of applications. The production line assembles bearing rolls, inner bearing ring and outer bearing ring to complete the bearing as a whole. Along this process, the bearing components are manufactured and finished by means of different processes like machining, hardening, grinding and honing. In this thesis, the honing process of the outer bearing ring is analyzed. A CNC machine performs the honing process. The CNC machine is a multi-axis CNC among which X1, Y3 and Y6 axis are used for the honing process. Figure 1.2 shows a schematic of the axis position in the CNC machine. This process runs for 24 hours and different variants of the outer ring are manufactured during this period.

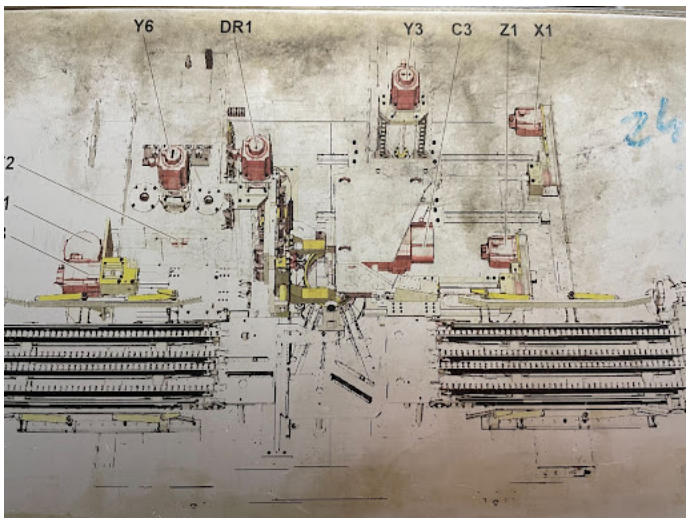

Figure 1.2: Schematic of CNC machine (Honing Process)

Siemens Edge device is installed on this CNC machine to extract the axis movement data. The data collection process is performed during the manufacturing process and the data is collected for all the manufactured rings. The Edge device captures the axis movement along with its respective torque values and provides it in the form of a JSON file. A single file contains data of all the three axes involved in this

process.

After the honing process, the ring goes under quality check before it is supplied for final assembly. The quality report for every ring is documented. The faulty rings are represented with Fault Code 1 and the non-faulty rings are represented with Fault Code 0. The faulty rings are again examined by the operator and after the final check, the ring is discarded or sent back to the assembly line. This process is critical and consumes some time. In this case, the problem addressed is the time-consuming quality process as well as pinpointing a trend in miscalibration of the CNC machine. The data from the edge device will be used to analyze any trends and provide real-time quality support and suggest any miscalibration trends by developing an ML model suitable for this case.

1.4 Limitations

This thesis deals with the data from only the above described machines for both manufacturing cases, therefore the results are only valid and specific for those machines. The data does not reflect real-time manufacturing conditions. The deployment stage of the model in both the cases is not considered due to the thesis time period and also this decision lies with the internal management of Manufacturing Company A and B. In order to conduct data analytics and ML modelling, PyCharm community version is used over the professional version due to the unavailability of license. This reduces the ease of extraction of graphs and results.

2

Theoretical Background

This chapter is the summary of the literature review carried out for performing this thesis which will also give the reader the necessary terminologies, concepts and basic knowledge for a better understanding of this thesis's scope. Therefore, this chapter is organized in a such way that readers can get a nice reading experience going from the broad to narrow aspects.

2.1 Industry 4.0 and Digitalization

The industry is the domain where the material goods, tools, clothes and many more things are manufactured. Technology played a crucial role in revolutionizing the manufacturing process over the period. These technology paradigm shifts are classified as Industrial Revolutions (Lasi et al., 2014). The first Industrial revolution was in the 1780s which is based on Mechanization driven by steam power. Before the first Industrial revolution, components were manufactured manually. The second industrial revolution was in the 1870s which is based on assembly lines and electricity. The third industrial revolution was in the 1970s which is based on electronics and information technology (M.-X. Lee et al., 2017).

Currently, the fourth Industrial Revolution which is also termed 'Industry 4.0' revolves around Smart Manufacturing as its main theme along with Smart working, Smart Supply chain and Smart products. Some focus concepts of Industry 4.0 are digitalization, integration of product life cycle and supply chain to the factory and the products produced (Frank et al., 2019). As per Gadre and Deoskar, 2020, some of the benefits of Industry 4.0 are reduction in production cost, reduction in labor cost, reduction in wastage, customizing the products at low cost, faster delivery time, prediction of equipment failures and finally increase in productivity.All these can be achieved by adopting various technologies which help in analyzing the gathered real-time data. The base technologies for Industry 4.0 are the Internet of Things (IoT), Cloud computing, big data and analytics (Frank et al., 2019). IoT ensures to have an interaction between the physical world and the virtual world where people and things are connected anytime and at any place. Cloud computing helps in connecting various computers and servers which can be personalized and can be easily assessed from anywhere in the world. The enormous data which is generated through various sensors in the digitalization phase is termed Big Data. Big Data gives many predictive and prescriptive insights, which can be drawn with the help of various analytic tools (Lampropoulos et al., 2019). AI is the domain

where research activities like robotics, ML, image processing etc take place. The integration of AI with the base technologies of Industry 4.0 will help the industries to run more efficiently (J. Lee et al., 2018).

2.2 Artificial Intelligence in Manufacturing Domain

There are two main types of manufacturing industry: discrete industries, which include machinery and equipment manufacturing; and process industries, which are represented by important raw material industries, such as the petrochemical, metallurgy, building material, and energy industries. Discrete manufacturing is a physical process, and its products can be counted individually. Therefore, it is easy to digitalize the manufacturing process, to emphasize personalized needs and flexible manufacturing (Yang et al., 2021).

The developments of industrialization have led to the optimization of production in real-time. It has become important to integrate AI with production to detect faults and irregularities without wasting much time. Irregularities and faults are inevitable in mass production and so companies that do not use AI spend more time detecting faults by browsing through thousands of product images (Zeba et al., 2021).The essence of industrial AI is to combine general AI technology with specific industrial scenarios in order to achieve innovative applications, such as design model innovation, intelligent production decision-making, and optimal resource allocation (Yang et al., 2021).

Companies collect huge amounts of data from different sources, such as sensors in machines, production lines, manufacturing execution systems, enterprise resource planning systems, systems outside of the production (customer feedback, supply chain), and other different purposes. Analyzing this data builds a competitive advantage and generates new products and services. Countries around the globe are implementing strategies and initiatives to keep pace with change, thanks to the rapid development of innovation and the digitalization of manufacturing (Zeba et al., 2021). Industrial AI grants an industrial system the capabilities of self-perception, self-learning, self-execution, self-decision-making, and self-adaptation, allowing it to adapt to a complex and changeable industrial environment and complete diversified industrial tasks; this ultimately improves production efficiency, product quality, and equipment performance (Yang et al., 2021).

Currently, there is a communication gap between all the entities in the production chain. The integration of Cyber-Physical Systems (CPS), IoT, Internetwork Operating System (IoS), Cloud Computing, Edge and Fog Computing, and other innovative technologies will enable end-to-end communication between production resources, production systems, and all supply chain stakeholders (Zeba et al., 2021). The manufacturing sector has the perfect fit for AI implementation. AI in manufacturing setup is being experimented and implemented in the areas like quality

assurance inspection, machinery maintenance, forecasting and real-time monitoring (Kehayov et al., 2022).

2.3 Predictive Maintenance

In any industry, maintenance is a very critical activity as it prolongs the machinery life, increases the machinery availability time and ensures everything goes as per the production plan. Also, the machinery conditions play a role in the product quality in the end (Kumar et al., 2013). In other words, improper maintenance of machinery will result in defects in the products and thus decreases the productivity of the manufacturing process. In addition to that, maintenance can play a crucial role in this competitive market by influencing quality and performance (Ran et al., 2019). Therefore, it is necessary to perform maintenance activities for the machinery. Over a period of time in the literature, different researchers grouped the maintenance strategies with different nomenclatures. Following is one type of nomenclature of the maintenance strategies (Susto et al., 2015):

- **Run-to-Failure (R2F)**: It is also called Corrective Maintenance. It is one of the basic maintenance strategies where maintenance activities take place only after equipment stops working. It is the least efficient strategy as the breakdown occurrence is not foreseen and an unplanned production stop has occurred. Due to this, there is an increase in the direct costs of the process.
- **Preventive Maintenance**: It is also called Time-based maintenance or Scheduled maintenance. In this strategy maintenance activities are carried out as per a fixed planned maintenance schedule. But it should be noted that at times performing the maintenance activities when not required will increase operating costs and wastage of resources.
- **Predictive Maintenance (PdM)**: It is the strategy where predictive tools are used to plan maintenance activities. Based on the historical data an early detection of failure is possible. In other terms, it can be said that it is based on continuous monitoring of the machinery.

As explained in Section 2.1, Industry 4.0 is the next big thing in industrialization. Where the integration of physical systems and digital systems takes place. Also, as said in section 2.2 there is a possibility of generating huge data in the industry which can be used in many areas. One such area is prognostics and health management (Çınar et al., 2020). In the form of PdM, the health of the machinery can be monitored to appropriately plan the maintenance activities. It is said that by adopting PdM, the maintenance costs can be reduced by 25%-35%, breakdowns can be eliminated by 70%-75%, breakdown time can be reduced by 35%-45% and finally, it results in an increase in production by 25%-35% (Montero Jimenez et al., 2020). Following are some of the single-model approaches of the PdM (Montero Jimenez et al., 2020):

- **Knowledge-based models**: These are the models that are made from prior experience. The experience here can be rules, facts or historical cases. The conclusions of these experiences can be automated by using some computational intelligence techniques. But it should be noted that sometimes only

experience is not enough to obtain accurate knowledge.

- **Data-driven models**: Due to the increase in the availability of huge data, this model is very popular now. This data can be used in studying components and machinery depletion.
- **Physics models**: These models are made with the help of physics and mathematics. To study the depletion behaviour of the components or machinery, simulations of the models are being performed.

Sometimes a combination of the above models is used to achieve better results in solving the complex systems (Montero Jimenez et al., 2020). This thesis is focused on a Data-driven model, so section 2.3.1 explains this approach in more detail.

2.3.1 Data-driven Approach

This approach has gained immense popularity and importance mainly due to the presence of huge data coming from the technical systems in the machinery and the computational power. The machinery is equipped with different kinds of sensors to extract various information and parameters. This will be very useful in knowing the current health status of the machinery or system and studying the degradation of the machinery. Statistics and Probability play a vital role in Data-driven models (Montero Jimenez et al., 2020). The data-driven models can be further classified into the following categories (Montero Jimenez et al., 2020):

- **Statistical models**: These models are typically used for finding the expected remaining life of the technical systems and the current degradation of the same. This is carried out by having a comparison of the present behaviour of measured random variables to the known behaviour which is depicted by a series of data. These models are generally adopted in multi-model approaches. Some approaches to these models are Regression analysis, Autoregressive and Bayesian.
- **Stochastic models**: These models are typically used for degradation modeling and remaining useful life estimation. These models are the probability models that focus on the evolution of random variables over time. Some of the main Stochastic processes which are the building blocks of these models are Gaussian, Markov and Levy processes.
- **Machine Learning models**: These models are built with the help of specialized learning algorithms from the data collected. These models are adopted especially to capture complex relationships among data which is tough to capture with the help of any other model or approach. It can be said that they play a crucial role in the PdM domain.

This thesis is based on the ML model of the Data-driven approach, section 2.5 explains this model and its classification in depth.

2.4 Review of Predictive Maintenance with Edge Intelligence

In this section, the implementation of Edge Intelligence in a general setup is explained and further how it is being used in the companies associated with this thesis is portrayed.

2.4.1 Edge Intelligence in General Setup

IIoT utilizes IoT devices and sensors to monitor machines and environments to ensure optimal performance of equipment and processes. PdM which monitors the health of machines to determine the probable failure of components is one IIoT technique that is receiving attention lately. As explained in Section 2.2 and section 2.3 to achieve effective PdM, massive amounts of data are collected, processed and it can ultimately be analyzed by ML algorithms (Hafeez et al., 2021). This has led to the implementation of edge devices in which both, sensors and intermediate nodes can process data. An edge device is a device that collects data or signals similar to sensors but has additional data processing capabilities. Edge Computing (EC) provides opportunities to reduce data transmission costs and increase processing speed. In a manufacturing setup, real-time data processing to monitor the manufacturing process, quality of the manufactured parts and part performance are of utmost importance. This means the process of decision support must be carried out at the edge. This will help in achieving zero-defect manufacturing. It can also be said that Edge Intelligence supports sustainable development (Kubiak et al., 2022).
There are three main techniques that utilize the EC paradigm to perform ML and data processing on intermediary nodes. These techniques are categorized according to where data processing occurs: Device and Edge, Edge and Cloud and Device and Cloud (Federated Learning).

- **Device and Edge**: In this setup, the data is collected by the edge device from the source and some preprocessing steps are performed on it. The preprocessing includes cleaning of the data and data reduction techniques. Further, the data is transferred to the cloud for model development. In the cloud, feature extraction is performed and further the ML model is developed and deployed in the cloud itself. The decision-making activities are performed in the cloud. It leads to delays in decision-making activities in a critical situation which is very ineffective. The main use of the Edge device in this setup is to collect larger sets of data and perform certain preprocessing steps which cannot be performed using sensors. Figure 2.1 represents the link between source, edge device and the cloud.

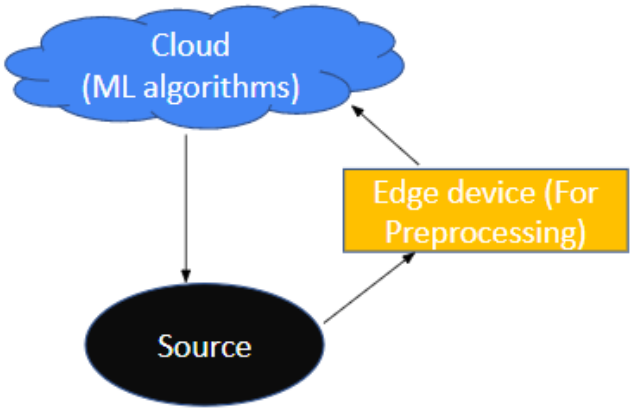

Figure 2.1: Device and Edge setup

- **Edge and Cloud**: In this setup, data is collected from the source by the edge device and transferred to the cloud after performing data cleaning and data reduction steps. The model is trained in the cloud and deployed on the edge device. This sets up local decision-making support for the critical activities. In the edge layer, sensors, terminals, and edge connection devices have the capacities for data analysis and processing, which support EC (Qi and Tao, 2019).The architecture requires fewer communication resources as well as less burden on cloud resources. If designed to work independently from the cloud, techniques can even work when there is no connection at all. The edge offers more context awareness as compared to cloud-based systems. With this architecture, meeting real-time requirements is possible. In the IIoT context, for example, real-time anomaly detection is very important to avoid a catastrophic situation. It also meets security and privacy concerns as edge locally processes data (Hafeez et al., 2021). It is worth noting that the ML model building and training take place in the cloud and then the model is pushed to the edge gateway. So, re-training of the model requires the new data to be pushed to the cloud and the model is deployed to the edge device after re-training the model. Figure 2.2 represents the link between source, edge device and the cloud.

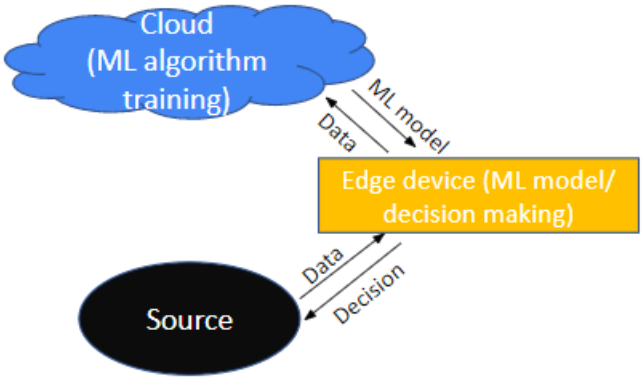

Figure 2.2: Edge and Cloud setup

- **Device and cloud**: In this setup, the ML model is developed and deployed directly on the edge device. The model is continuously trained on the raw signals coming from the source. The updated model is sent to the cloud. The model in the cloud is used globally to deploy directly on the edge device or used directly in decision support tasks. This setup meets privacy requirements because raw data are not leaving the source. It minimizes the communication burden as only the model updates are forwarded, not the raw data. The approach relies on having sufficient computation resources to train models on the edge device (Hafeez et al., 2021). Figure 2.3 represents the link between source, edge device and the cloud.

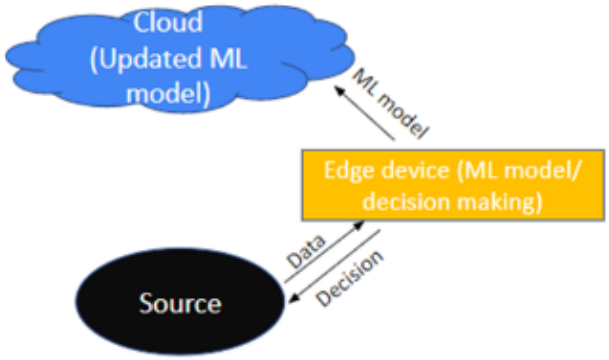

Figure 2.3: Device and Cloud setup

Unfortunately, deploying ML in an IoT system faces challenges due to constraints of the IoT system. For example, if ML is implemented in the cloud, real-time local decision-making is almost impossible to achieve due to underlying limited bandwidth connectivity between sensing nodes and the cloud. To address the problem, ML can be deployed on the device. However, the limited computing capacity of the sensing nodes is a major challenge. Therefore, a hybrid architecture to implement computation-intensive tasks such as training on the cloud and deploying models for prediction on the sensing node has emerged (Qi and Tao, 2019).

In this thesis, an edge device is used in a hybrid setup where processed data from the edge will be transferred to the model for local decision-making support. The key purpose of the Edge device in this case is to analyze the machine condition, therefore the measurement characteristics represent the data related to condition monitoring of the machine.

2.4.2 Measurement Characteristics in Edge Device

The edge device provided by Siemens is being used by the Manufacturing company A and Manufacturing company B. In addition to the basic services of the edge device, Manufacturing company A also uses certain features of the edge device to analyze specific conditions on the CNC machine. These features are independent measurement characteristics of the CNC machine and the explanation for the measurement

characteristics provided my the corresponding stakeholders are given below. The in-depth test details and procedures are not disclosed due to confidentiality purposes.

- **Equability Test**: In the equability test, an axis is moved at a constant velocity over the defined measurement path. The alternating components of the load-side force with respect to the axis positions are determined from the measured motor torque. The characteristic is the maximum force during travel in both directions. The maximum force increases due to insufficient lubrication, mechanical damage on the axis, jammed cover segments, or chip contamination. This leads to reduced workpiece quality and damage to the machine itself.
- **Friction Test**: The friction test provides measurement results for dry friction, viscous friction and friction distribution. The friction distribution provides information on how the axial friction is distributed between the spindle nut or the guidance. In this way, it is possible to estimate which component of the drive train is responsible for the increase in friction as total friction increases. If the increased friction results in increased compression between the measuring systems, then the friction on the guide side has increased. Otherwise, the friction has increased near the motor - at the spindle nut or upstream gear. An increase in friction distribution on the spindle nut is caused by a mechanical defect in the spindle nut or insufficient lubrication of the spindle nut. An increase in friction distribution on the guidance is caused by a mechanical defect on the guide carriage running surface, insufficient lubrication of the guide or insufficient lubrication of the cover. This leads to positioning errors during any machining run.
- **Backlash Test**: Backlash is a fault in positioning that occurs when the direction of force is reversed. It is caused by play and low levels of stiffness in the drive train. The backlash also affects bidirectional repeat accuracy. This is caused by wear in the guide grooves of the ball screw or spindle nut or by plastic deformation of ball bearings in the ball screw. This leads to positioning errors, surface defects and vibration of the machine during traversing movements with fast and frequent changes of direction.
- **Stiffness Test**: Stiffness describes the overall stiffness of the drive train between the two measuring systems and can also be understood as the axial stiffness of the entire drive train of an axis. In a ball screw, overall stiffness decreases as the distance from the locating bearing increases since the stiffness of the ball screw (tension/compression and rotation) decreases with increasing length. Stiffness does not normally increase. The stiffness of the axes of a machine tool can decrease over time due to diminishing ball screw pre-tension, wear on the ball bearings in the guide and damage to the bearing. This leads to reduced workpiece quality and a fall in the lowest natural frequency of the drive train.
- **Signature Test**: The signature indicates periodic synchronous positioning errors due to location-dependent faults in the drive train. The signature is ascertained from the frequency ranges of several different constant speeds. If an order occurs at least three different speeds, it can be excluded as a fault, e.g. due to an excited natural frequency. The order is adopted as a

parameter. Defective components in the drive train can be identified based on the signature. In such a case the comparison of the signature measurement with the reference measurement shows a new order, which was not present in the reference measurement. This is caused by bearing damage, damage to the ball screw or loss of tension in a toothed belt. This leads to surface defects due to vibrations and positioning errors.

Among these five tests, data corresponding to only three tests (Friction test, Backlash test and Stiffness test) were analyzed and ML models were developed to analyze these tests. All the tests were not analyzed due to limited time-frame and knowledge about the tests.

2.5 Machine Learning Algorithms

ML has become one of the important aspects of AI and information technology. In addition to that, as said in section 2.3, it is one of the powerful tools in the PdM domain. But it is important to know that a model can be trained in different ways with the help of different algorithms based on the type of input data and interactions with the environment (Kaur and Jindal, 2016). The ML models can be categorized as per their learning styles. Following are the different ML model categories

- **Supervised Learning**: In these ML models, values are predicted as approved or not approved with the help of a learned target function. Generally, the data is split into training dataset and test dataset where the training dataset is used to train the model and the test data is used to check the accuracy of the model. Some of the typical methods of supervised learning are Classification and Regression.
- **Unsupervised Learning**: In these ML models, there is an absence of labeled data or class labels of training data. The model is being trained by the algorithm to find the patterns and group them in clusters for further interpretation. Clustering is one of the widely used methods under unsupervised learning.
- **Reinforcement Learning**: In these ML models, the experience is used to train the model instead of examples. Adding to that, the algorithm learns about its decision by choosing the actions on each data point.

There are many ML algorithms for each category which are explained above. An overview of some popular algorithms is shown in the Figure 2.4. It should be noted that each algorithm has its own advantages, limitations and application areas. Also, it is necessary to select the right algorithm as it helps in understanding the role of the input data, choosing the model preparation process and getting the desired results (Kaur and Jindal, 2016).

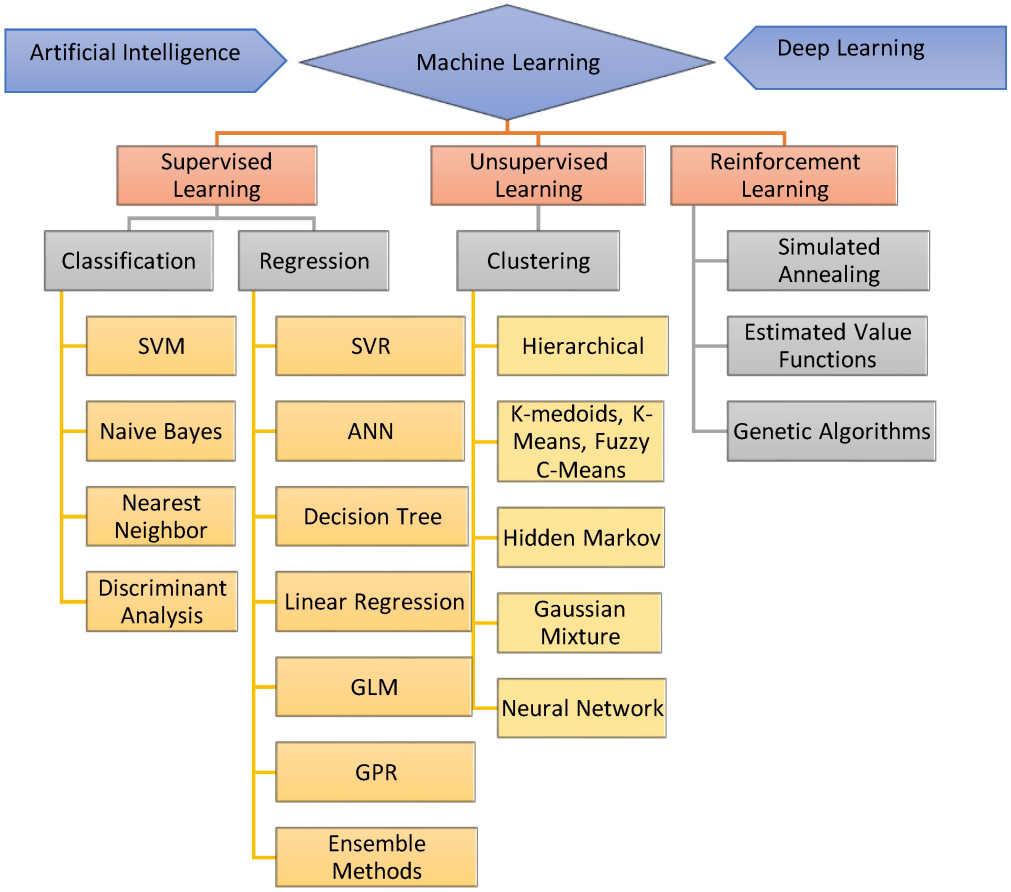

Figure 2.4: Overview of Popular ML algorithms (Çınar et al., 2020)

2.5.1 A Brief Overview of Used Machine Learning Algorithms in This Thesis

The proposed approach is to develop an individual ML model for each test considered in this thesis for Manufacturing company A as described in section 2.4 to easily distinguish the results as the tests working conditions and focus variables are different. Also, a separate ML model is made for Manufacturing company B in accordance with its problem as described in section 1.3. The algorithms used for this thesis are explained in the following sections.

2.5.1.1 Gaussian Naive Bayes

Naive Bayes is a supervised ML classification algorithm which is simple and efficient. It is based on the Bayes theorem and possesses high functionality (Majumder, n.d.). But it is loosely based on an assumption of conditional independence between the attributes. Gaussian Naive Bayes (GNB) is out of the many algorithms proposed to improve its efficiency from various viewpoints. In GNB, it is assumed that the continuous values of a class resemble a Gaussian distribution (Jahromi and Taheri, 2017). This algorithm fits the model with the help of the mean and standard deviation of the points of an individual label (Majumder, n.d.). The probability of

estimating the continuous data is given by the following equation (Jahromi and Taheri, 2017):

$$p(x_i|c) = \frac{1}{\sqrt{2\pi\sigma_{c,i}^2}} \exp^{(-\frac{(x_i-\mu_{c,i})^2}{2\sigma_{c,i}^2})} \tag{2.1}$$

2.5.1.2 Decision Tree

Decision Tree (DT) is a supervised ML algorithm whose structure resembles an actual tree. The DT algorithm contains a root node, branches and leaf nodes which are the same as the root, branches and leaves of an actual tree. The root node is the uppermost node of the DT algorithm, it is also called the parent of all the nodes. Each branch of DT represents a decision or rule which at the end gives out a leaf node as an outcome. This outcome can either be a categorical or numerical value (Patel and Prajapati, 2018). Figure 2.5 represents a typical structure of DT algorithm. So, it can be said that the DT algorithm learns through a series of if-else clauses where a complicated decision-making mechanism is made into simplified decisions and finally associate the labels to the data points (Çınar et al., 2020).

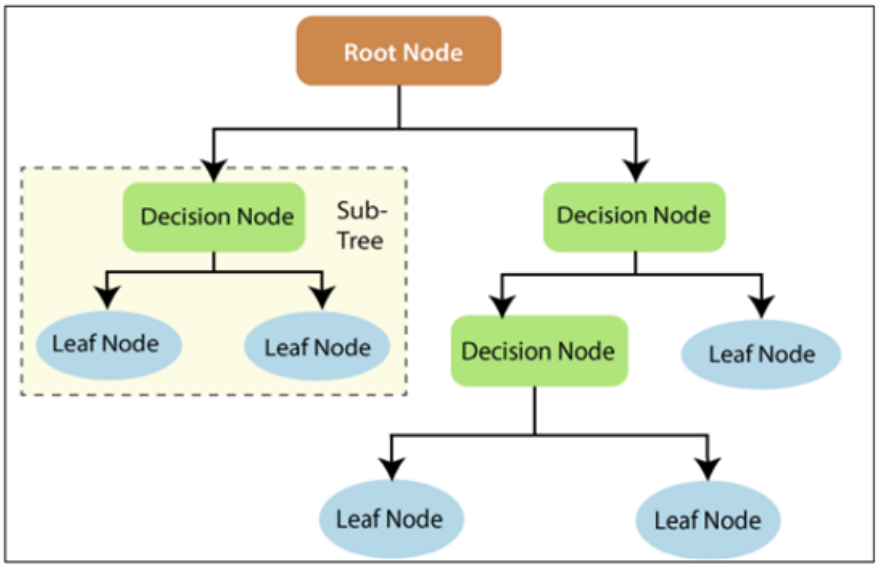

Figure 2.5: Structure of DT algorithm (Charbuty and Abdulazeez, 2021)

DT algorithm is widely used for grouping purposes. It is easy to interpret and can be adopted for both clustering and regression problems. It is widely used in areas such as medical diagnosis, voice recognition, character recognition, banking, planning logistics, finance etc (Charbuty and Abdulazeez, 2021).

2.5.1.3 Random Forest

Random Forest (RF) is a supervised learning algorithm that can be used in both classification and regression related problems. In simpler terms, RF is an ensemble of many DTs which eventually increases the overall accuracy (Donges, 2022). In RF,

every tree gives a vote for each value on which class label it belongs. Finally, RF predicts by analyzing the votes from all the tress for a value, most likely considering the majority votes (Chaudhary et al., 2016). The advantages of RF are, not easily over-fitted, robust to noise, adds additional randomness etc. As RF merges many DTs it requires rigorous training.

2.5.1.4 RUSBoost

In a few ML problems, there are some instances where the data can be imbalanced. In other terms, it can be said that the dataset is skewed towards one class than the other. In such situations, the standard ML which affects the predicting accuracy of the minor class (Tanha et al., 2020). It is necessary to use an algorithm that is better at handling imbalanced data and maintaining good performance of the model.

Following are the two methods which are adopted for handling imbalanced datasets:

- **Data-level approaches**: In this approach, the rebalance takes place with respect to data distribution by resampling methods. Where majority class observations are decreased, or minor class observations are increased by extrapolation.
- **Algorithm-level approaches**: In this approach, the imbalanced data is adjusted by modifying the ML algorithm itself.

For this thesis one of the Data-level approach algorithm is used, which is "RUSBoost". This algorithm is chosen as it uses the under-sampling technique where the majority class observations are decreased. The advantage of this is it reduces the time required to train the data and avoids overfitting. In addition to that, the under-sampling techniques have less computational requirements and are more effective than others (Seiffert et al., 2008).

RUSBoost is a combination of random under-sampling and boosting techniques. In this algorithm, under-sampling is performed by excluding the minority class so that all the classes have some percentage of data. Later AdaBoost technique is applied so that the same training data is not repeated in every iteration (Tanha et al., 2020). Some of the advantages of the RUSBoost algorithm are that it is computationally less expensive, has lesser training time and provides better competitive performance (Seiffert et al., 2008). But it should be noted that there is data loss during the under-sampling phase. It can be used for both binary and multi-class classification problems (Tanha et al., 2020).

Generally, evaluation metrics are used to evaluate the performance of the built model. Various metrics can be used for this process. Section 2.6 gives deep insights into the available metrics and the metrics that are used for this thesis.

2.6 Evaluation Metrics

Evaluation metrics are a part of every ML pipeline. They provide information about the progress and efficiency of the model. The idea of building ML models works on a constructive feedback principle. After a model is built, feedback is received from metrics, and further improvements are made until desired results are achieved. An important aspect of evaluation metrics is their capability to discriminate among model results (Tavish, 2019). The predictive models are either a regression model (continuous output) or a classification model (nominal or binary output). The evaluation metrics used in each of these models are different.

In classification problems, two types of algorithms (dependent on the kind of output it creates) are used:

- **Class output**: Algorithms like Support Vector Machine (SVM) and k-Nearest Neighbor (KNN) create a class output. For instance, in a binary classification problem, the outputs will be either 0 or 1.
- **Probability output**: Algorithms like Logistic Regression, Random Forest, Gradient Boosting, Adaboost etc. give probability outputs.

In regression problems, there are no such inconsistencies in the output. The output is always continuous in nature and requires no further treatment (Tavish, 2019).

2.6.1 Evaluation Matrix in Classification Models

This section explains the matrices used in the classification problem to evaluate the performance and improve the model.

Confusion Matrix:
Confusion matrix gives a good overview of the predictions made by the model and how accurately it predicts the right class. It is a tabular visualization of the ground-truth labels versus model predictions. Each column of the confusion matrix represents the instances in a predicted class and each row represents the instances in an actual class. Confusion Matrix is not a performance metric but act as a base for other evaluation metrics (Bajaj, 2022).The cell allocation is presented in Figure 2.6. They are as follows:

- **True Positive(TP)**: It signifies how many positive class samples the model predicted correctly.
- **True Negative(TN)**: It signifies how many negative class samples the model predicted correctly.
- **False Positive(FP)**: It signifies how many negative class samples your model predicted incorrectly. This factor represents **Type-I error** in statistical nomenclature.
- **False Negative(FN)**: It signifies how many positive class samples your model predicted incorrectly. This factor represents **Type-II error** in statistical nomenclature.

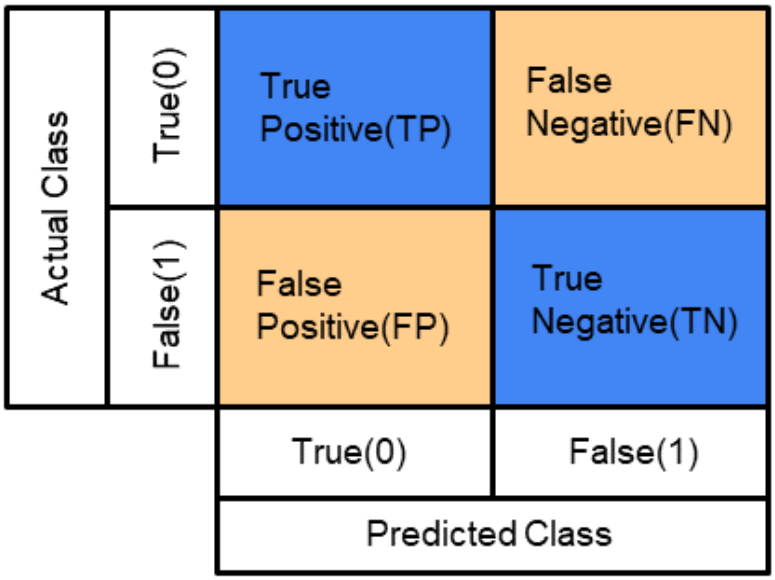

Figure 2.6: Confusion Matrix (Draelos, 2019)

Accuracy:
Classification accuracy is the simplest metric used for model evaluation. It is defined as the number of correct predictions divided by the total number of predictions, multiplied by 100 (Bajaj, 2022).

Precision:
Precision is the ratio of true positives and total positives predicted:

$$Precision = \frac{TP}{TP + FP} \tag{2.2}$$

The precision metric focuses on Type-I error (FP). A precision score of 1 signifies that the model did not miss any true positives, and is able to classify well between the labels. It cannot measure Type-II error, which is false negatives. A low precision score (<0.5) means the classifier has a high number of false positives which can be an outcome of an imbalanced class or untuned model parameters (Bajaj, 2022).

Recall:
A Recall is essentially the ratio of true positives to all the positives in the ground truth.

$$Recall = \frac{TP}{TP + FN} \tag{2.3}$$

The recall metric focuses on type-II errors (FN). Recall score of 1 signifies that the model did not miss any true positives, and is able to classify well between the labels. It cannot measure the existence of type-I error. A low recall score (<0.5) means the classifier has a high number of false negatives which can be an outcome of an imbalanced class or untuned model parameters (Bajaj, 2022).

F1-Score:
It is difficult to compare two models with low precision and high recall or vice versa.

So, to make them comparable, F1-Score can be used (Narkhede, 2018).The F1-score metric uses the harmonic mean of precision and recall to analyze the model. The formula for calculating F1 score is:

$$F1 - Score = \frac{2}{\frac{1}{Precision} + \frac{1}{Recall}} \tag{2.4}$$

A high F1 score signifies a high precision as well as high recall. It presents a good balance between precision and recall and gives good results on imbalanced classification problems. A low F1 score signifies nothing, it only talks about performance at a threshold. With low F1 score, it's unclear what the problem is (low precision or low recall), and whether the model suffers from type-I or type-II error (Bajaj, 2022).

3
Methodology

This chapter presents the methodology considered and adopted for this thesis. It aims to help the reader understand how the steps of this methodology were adapted and implemented according the requirements of two manufacturing cases including the experiment and analysis details and the utilized tools during the analysis.

For the thesis, a comparative study was carried out for comparing both case studies. Whereas a different structured work procedure or methodology was used as a foundation for analysing the case studies. This was done for performing a better-quality assurance analysis, reducing lead time and having a better communication structure. The Cross-Industry Standard Process for Data Mining (CRISP-DM) is used for adoption for case study analysis.

3.1 CRISP-DM in Manufacturing Domain

CRISP-DM can be used anywhere irrespective of its sector. In other terms, it can be used by anyone repeatedly irrespective of the work situation (Wirth and Hipp, 2000). Figure 3.1 depicts the general CRISP-DM model. This model is an iterative model where some steps should be repeated till the results are in line with the business objectives.

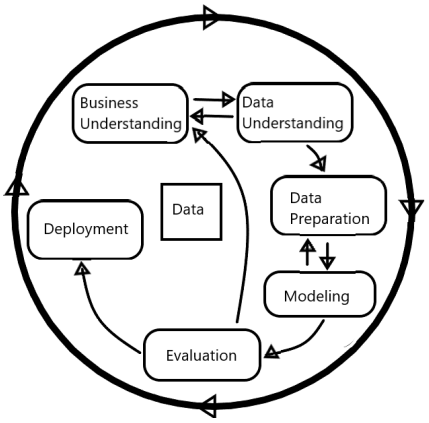

Figure 3.1: CRISP-DM Methodology

Over the past, CRISP-DM has been used for various problems in the Manufacturing domain. It has been used by Maintenance and Repair organizations where mainte-

nance tasks were improvised for increasing the availability of aircraft during the peak season. It was also used in predicting the failure of the aircraft components (Pelt, Maurice et al., 2019). Many theses were also performed with the help of CRISP-DM. Some of them focused on, using collected data to improve the production line performance from an economic perspective, developing a data-driven approach to quickly identify the pneumatic leakage in a production process and analyzing the alarms triggering in the manufacturing industry (Todorovac and Wiking, 2021)(Lené and Rajashekarappa, 2021)(Vasudevan and Duan, 2021). There were no major drawbacks presented in the above-said examples except the fact that some expressed the unclearness of the Data acquisition phase. The CRISP-DM has ensured to have a structured guideline in solving the above-said problems. The detailed version of the adopted CRISP-DM methodology for the case studies analysis in this thesis is explained in section 3.2.

3.2 Adopted CRISP-DM Methodology

The adopted version of the CRISP-DM methodology for the case studies is explained in the following sections. It should be noted that for both the case studies the data acquisition phase is not considered as the data was already acquired by the companies.

3.2.1 Business Understanding

In this initial phase, the objectives are set from a business point of view by understanding the requirements of the project. Along with this, the situation was assessed in terms of resources, assumptions, risks, costs, and benefits. This was assessed with the help of interactions with the necessary corresponding stakeholders. After assessing all the necessary details, the information was used to transform the requirements into a data-driven problem. In addition to that scope of the thesis was defined and a project plan was made. This is a crucial phase as it decides the success of the project.

3.2.2 Data Understanding

In this phase, the collected data was explored to find the familiarities and the uniqueness in it to accomplish the thesis objectives by considering the business objectives identified in the previous step. In addition to that data was being described in terms of data format. A detailed data quality report was made for identifying the data problems which are further explained in section 3.2.2.1. An exploratory data analysis (EDA) was carried out for identifying the relationships among the data which is further explained in section 3.2.2.2.

3.2.2.1 Data Quality

Data quality measures the condition of the data available on various parameters. It is important to measure the quality of data to analyse and understand if the data

justifies its purpose of use. The data quality was analyzed with the help of various quality dimensions and elements designed by Cai and Zhu, 2015.

3.2.2.2 Exploratory Data Analysis

EDA is used for analyzing the data by descriptive data summarization and visualization. EDA is used to find the hidden patterns or relationships of the data, outliers, generate hypotheses and other anomalies. Descriptive data summarization consists of features like mean, median, standard deviation, pairwise correlations etc. Visualization involves representing the data in a visual format with the help of various graphs for better understanding and analysis ("Exploratory Data Analysis", n.d.).

As per this thesis aim, the descriptive data summarization was not of effective use. But visualization was extensively used in this thesis for data analysis. Out of the various data visualization tools, Python was used with histograms, box plots and scatter plots as some of the main visualization techniques for this thesis. Python was chosen for the ease of its usage, open-source language and the ability to handle the JSON files a bit better than the other software packages. PyCharm was used for executing the python programs as it was compatible with system use and beginner-friendly.

3.2.3 Data Preparation

The data preparation phase covers activities like data selection, data cleaning, data construction, feature extraction and building the final dataset to feed the model. In the data analysis process major time (around 80%) is spent on data cleaning and preparation stage (Dasu and Johnson, 2003). A dataset is a set of columns and rows where the data can be either quantitative (numbers) or qualitative(strings). Every value in a dataset is associated to a variable and an observation (Wickham, 2014).

In this thesis, different approaches have been taken for Manufacturing Company A and Manufacturing Company B to prepare the data depending the explored data quality problems and the requirement of the case study. These approaches are explained in the following subsection.

3.2.3.1 Manufacturing Company A

Manufacturing Company A deals with the CNC machine and the data collected is specific to the tests explained in section 2.4.2. The data includes the axis movement values along a linear or rotational direction or both depending on its degree of freedom. Along with the movement values it also includes probe counter numbers (data collection points), the torque values of the drive during these movements, the desired position of the axis, G-Codes for the movement and the Call command for the tests. The acquired data are named ENC1(Encoder position on the drive), ENC2(Encoder position on the guideway), Torque (Torque on the drive) and DES (Desired position of command axis). Every axis has a different file which includes the data for all the tests performed. Initially, the data was extracted from the JSON

files and made as a Python dataframe. The headers in the files were not always in a sequence, so a generalized code was developed to detect the headers and then assign them to the respective columns in the dataframe. Further, this dataframe was split into five different dataframes as per the tests (Equability, Friction, Backlash, Stiffness and Signature). This split was performed on the basis of the call command in the dataset for every test. After a comprehensive EDA, the data points from all the tests were trimmed to remove some redundant data. Trimming was performed as per the G-codes in the dataset. Later, the tests were analyzed individually for their relevance and effect on the machine condition. The terminologies used in the data preparation and feature generation stage for every test are explained below:

- **ENC1 − DES**: It is the difference between the actual position of the drive and the desired position of the drive at a specific instance.
- **ENC2 − DES**: It is the difference between the actual position of the guideway and the desired position of the guideway at a specific instance.
- **ENC2 − ENC1**: It is the difference between the position of the guideway and the position of the drive at a specific instance.
- **Torque**: It is the torque generated on the drive at a specific instance.

The data was further used to prepare specific features and it was classified into different class as per the risk levels. Only three tests were considered for model preparation due to limited time-frame and lack of knowledge about other tests. Every test had its own features and risk levels which are explained below:

- **Friction Test**: Friction has two parts, dry friction and viscous friction. Dry friction does not rely on speed whereas viscous friction relies on speed as explained in section 2.4.2. The Friction test can fail either on the drive side or on the guide side in a CNC machine. To tackle this, movement of both, the drive and the guide were taken into consideration along with the torque values. The friction dataframe was divided into two dataframes, one for the drive and the other for the guide. The variables to analyze the friction test on the drive were ENC1-DES and Torque. The variables to analyze the friction test on the guide were ENC2-DES and Torque. The data available lacked faulty data as well as quality report since no parts were manufactured during the tests. This had an effect while classifying the data as risky or not. Therefore, to classify the data, it was assumed that the highest value of ENC-DES lies in the risk zone. Positive and negative friction were considered, and the data was classified into four classes, ranging from lowest risk to highest risk. The classes were prepared so that the data is not very skewed and the model can learn the patterns of all classes.
- **Backlash Test**: Backlash is basically slippage in the movement during changing directions. It can be detected on the guideway of the CNC machine. To analyze backlash, movement on the guide was considered. The backlash dataset has only one variable which is the modulus of ENC2-ENC1. The data available lacked faulty data as well as quality report since no parts were manufactured during the tests. This had an effect while classifying the data as risky or not. As per "Anti Backlash for CNC", n.d. backlash test fails if the difference is above 0.005 inches or 0.127 mm. The existing data do not have any value

above 0.127 mm. If a class is set for values above 0.127, the model will not have any data of this class to train on. Taking note of this, three classes were formed, ranging from lowest risk to highest risk.

- **Stiffness Test**: Stiffness is observed on the bearings of the guideway. The stiffness is reduced due to defects in the bearings whereas stiffness usually do not increase as explained in section 2.4.2. To analyze stiffness, the difference between the movement on the guide and the drive was considered. This was formulated for finding the additional movement on the guide with respect to the movement on the drive. The stiffness dataset has only one variable which is ENC2-ENC1. The data available lacked faulty data as well as quality report since no parts were manufactured during the tests. This had an effect while classifying the data as risky or not. Therefore, to classify the data, it was assumed that the highest value of ENC2-ENC1 lies in the risk zone. Data was divided into four classes where one class defines an increase in stiffness while the other three define a decrease in stiffness from lowest risk to highest risk.

3.2.3.2 Manufacturing Company B

Manufacturing Company B deals with the honing process of the outer bearing ring. The data collected is the axis movements during honing of each ring. Along with the movement values it also includes the torque values of the drive during these movements and the desired position of the axis. The acquired data uses the following terminologies:

- **ENC_POS|1**: Encoder position of X1 axis.
- **ENC_POS|7**: Encoder position of Y3 axis.
- **ENC_POS|14**: Encoder position of Y6 axis.
- **DES_POS|1**: Desired position of X1 axis.
- **DES_POS|7**: Desired position of Y3 axis.
- **DES_POS|14**: Desired position of Y6 axis.
- **TORQUE|1**: Torque generated on X1 axis.
- **TORQUE|7**: Torque generated on Y3 axis.
- **TORQUE|14**: Torque generated on Y6 axis.

A single JSON file includes the manufacturing data of one outer ring along with its serial number and timestamp. The headers in the files were not always in a sequence, so a generalized code was developed to detect the headers and then assign them to the respective columns in the dataframe. Initially, data of all the outer rings were combined into one dataframe in Python. A comprehensive EDA was performed to understand the correlation between torque, encoder position and the desired position during the honing process. The data was reduced and the following variables were considered for the final dataset:

- ENC_POS|1 - DES_POS|1
- ENC_POS|7 - DES_POS|7
- ENC_POS|14 - DES_POS|14
- TORQUE|1
- TORQUE|7

- TORQUE|14
- DES_POS|1
- DES_POS|7
- DES_POS|14

A quality report for the manufactured rings was provided by Manufacturing Company B. This quality report included the serial number, timestamp and the fault code for each part where fault code 0 represented a non-faulty part and fault code 1 represented a faulty part. This data was merged with the machining data as per the serial number and timestamp. The final dataframe was classified into two classes with the help of quality data i.e., faulty or non-faulty. Further, feature extraction was performed on the above-mentioned variables, where minimum value, maximum value, mean, standard deviation, skewness and kurtosis were extracted. These features were ranked by using Fisher score method. Fisher score is most widely used algorithm for determining most discriminative subset of features (Vakharia and Kankar, 2016). After the feature ranking, top 10 and top 15 features were considered for the modeling phase.

3.2.4 Modelling

In this phase, the model was selected from various options available that meet the business objectives set in section 3.2.1. Later, a test design was made for building the model. Finally, the model was built by finetuning some parameters.

A similar test design was carried out for both the Manufacturing Company cases where the final dataset was primarily split into two datasets. The first dataset was used for building the model. This dataset was further split into training and testing data. The model was trained with the help of training data and was validated with the Test data. This split was done using a train-test-split function of the Python library sklearn model. This function was used by giving certain inputs to perform the split operation. Some of the important inputs given for this function were, the dataset of all independent variables, the dataset of only the dependent variable or the target class, test size that defines the proportion of the dataset to be considered as Test data. In this thesis, “0.2” was used as test size, which implies that 20% of the dataset was classified as Test data. The second dataset was used for the double validation with only the faulty data for Manufacturing Company B. Whereas for Manufacturing Company A, the second dataset classes were set by rule-base codes and then used for the double validation. Figure 3.2 depicts the systematic way in which data was handled during the modelling phase.

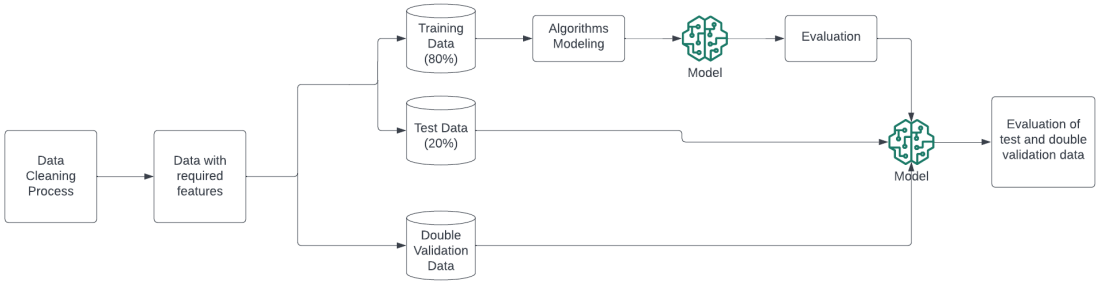

Figure 3.2: Data Flow during the modelling phase

3.2.5 Evaluation

In this phase, the models generated are evaluated with the help of evaluation metrics to understand if the model meets the business objectives of the organization. In addition to that, it can also be verified if the model meets the success criteria. For this thesis as explained in section 2.6, classification model evaluation metrics are used to check the performance of the models. Therefore, as explained in section 2.6.1, the Confusion matrix, Accuracy, F1-Score, Recall and Precision are considered for the final evaluation of the model. In addition to that as specified in section 3.2.4, certain data which was termed validation data was used for the preliminary validation of the models. Finally, model performance was evaluated with double validation data.

3.2.6 Deployment

This is the final phase where a report is made and presentations are given to the organization to clearly communicate the findings & knowledge gained through the project. In addition to that, the model built is implemented in the organization. For this thesis, deploying the model is not in the scope due to the limited time as well as the decision lies on the company's internal management. However, the findings are communicated to both the manufacturing companies in accordance with the better usage of Siemens Edge device data and reducing their problems as described in section 1.3. In addition to that, some recommendations will be given to the manufacturing companies on how they can deploy those models in their real production environments using Edge Intelligence for robust implementation of PdM.

4
Results

In this chapter, the results of both case studies are presented along with their comparative study. The results are presented for both cases as per the adopted CRISP-DM methodology steps as explained in section 3.2. However, as specified before, the deployment phase for both case studies is not considered in this thesis.

4.1 Case Study 1: Manufacturing Company A

4.1.1 Business Understanding

The business objective of this case was understood by having meetings with involved stakeholders. Following are the stakeholders involved in this case:

- Maintenance Engineer of Manufacturing company A
- Verification Engineer of Manufacturing company A
- Maintenance Reliability person of Manufacturing company A
- PACA Project academic partners
- Academic Supervisors

After a few meetings with the above stakeholders, the business objective was defined, which is to use the raw data coming from the Siemens Edge device for decision support in scheduling maintenance activities of a CNC machine. Currently, at Manufacturing company A, processed data of the edge device is used for checking the calibration of the CNC machine instead of raw data which increases the lead time of the process. With the business objective being defined, a problem description was made which was explained in section 1.3.

4.1.2 Data Understanding

4.1.2.1 Data Quality Report

The data files and other supporting documents were provided by Manufacturing company A and Siemens through a cloud platform called BOX which was very easy to access. But the data file formats are JSON files that are not so easily retrievable, unlike XML or TXT files. The time consumed to retrieve the JSON files to ease the accessibility of the data was high. The data is being collected from October 2021 on a frequency of once per week and uploaded in BOX mostly on Fridays and sometimes Mondays. The reason for collecting the data only one day per week is due to fact that it is been measured in a standstill production line. This thesis was

started in January 2022, until which the data was readily available.

Specification of Data names is given, for example, Axis names, Encoders, Probe counters, different Tests and G-Codes. The definition of the above data names is missing except for the tests where the description is vaguely given. The ranges of all the data values are missing which makes it difficult to comment if the given data exists in the range of known and accepted values. Data is considered credible as it comes directly from Manufacturing Company A. As per our understanding, the data audit is not being carried out. A standard format is not followed, as it was observed that the order of certain columns was not uniform in the header over time. As per the communication with the necessary stakeholder, the reasons behind this are unknown. However, some extra care was taken during the coding process to eliminate this issue.

The given data is raw data; therefore it is not formatted before uploading in BOX and as of now, the data is coming from only one source. It is also observed that data is consistent over time which was also verified by plotting the regression plots. The data corresponding to the Encoder 2 position for B-axis was missing but after the communication with necessary stakeholders, it was ignored as B-axis has only a motor measuring system and therefore there is no requirement for Encoder 2. Also, there were only three tests conducted for the B-axis instead of five because of the possibility of conduction of only those in a direct measuring system.

The presence of domain knowledge will make the given data understandable to some extent. Therefore, it is not so easy to judge if the data meets the needs or not. The classification of data as per axes is easy to understand as different axis data is saved as a separate individual file. But the classification of data corresponding to various tests is a bit difficult as all the various test data of an axis are in a single file. In addition to that, it was not so easy to transform this semi-structured data into a structured format.

4.1.2.2 Exploratory Data Analysis

The data exploration stage led to certain insights about the data provided from the edge device. As specified in section 4.1.2.1, it was found that the headers in the files were not in the same order throughout the data collection period. The X-axis in Figure 4.1 shows the order of headers in different months. To counter this situation, a general code was written which can extract the data even if the headers are not in sequential order. Further, box plots were extracted to find the variation in the data over the period. Figure 4.1 shows that there was very minor variation in the recorded values. This is a similar situation for all the axes. This validates the fact said by the stakeholders that the machine is in healthy condition to date.

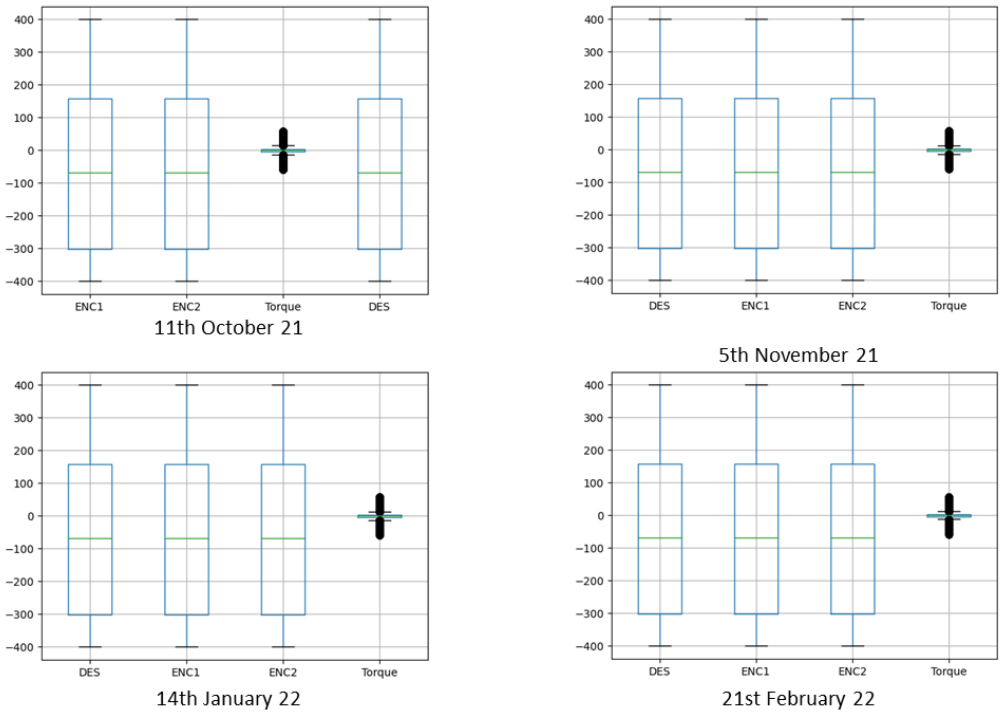

Figure 4.1: Box Plots for the X-axis over a period of time

Regression plots for all the tests and axes were extracted to understand the pattern of movement of every axis during different tests. Figure 4.2 shows the pattern of all the tests over a period of time for the X-axis. Regression plots of the Y, Z, A and B axis are shown in appendix A. It was observed that the pattern of movement in every axis is very similar over the period of time. So, it was assumed that the variation for all the axis is similar, which led to the decision of using the same limits in the final dataset for all the axis.

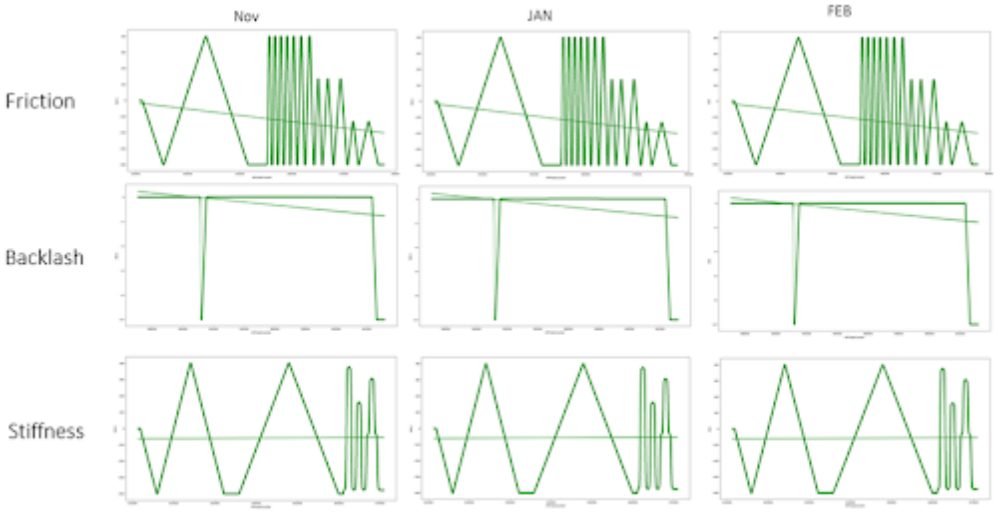

Figure 4.2: Regression Plots of the tests over a period of time

A comparative study between the torque and the encoder values was performed for the friction test. The Figure 4.3 represents the regression plots of torque and encoders of the X axis. Regression plots of the Y, Z, A and B axis are shown in appendix B. It is evident from the graphs that, the test was performed at different torque values to understand the effect of speed on friction. As per our observations, there are very minor variations due to friction since the machine is in healthy condition.

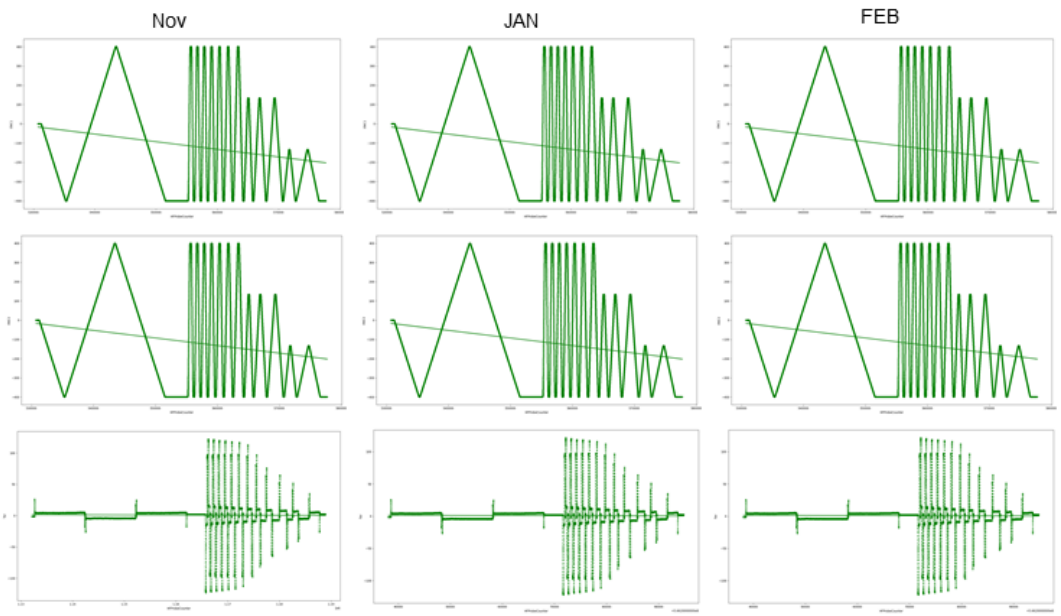

Figure 4.3: Encoder and Torque comparison for friction test over a period of time

4.1.3 Data Preparation

In the data preparation stage, the dataset of every test was trimmed to remove the movement of the axis which was not part of the test runs (movements to relocate the axis to start the next run), Figure 4.4 shows the regression plots of the dataset before trimming and after trimming.

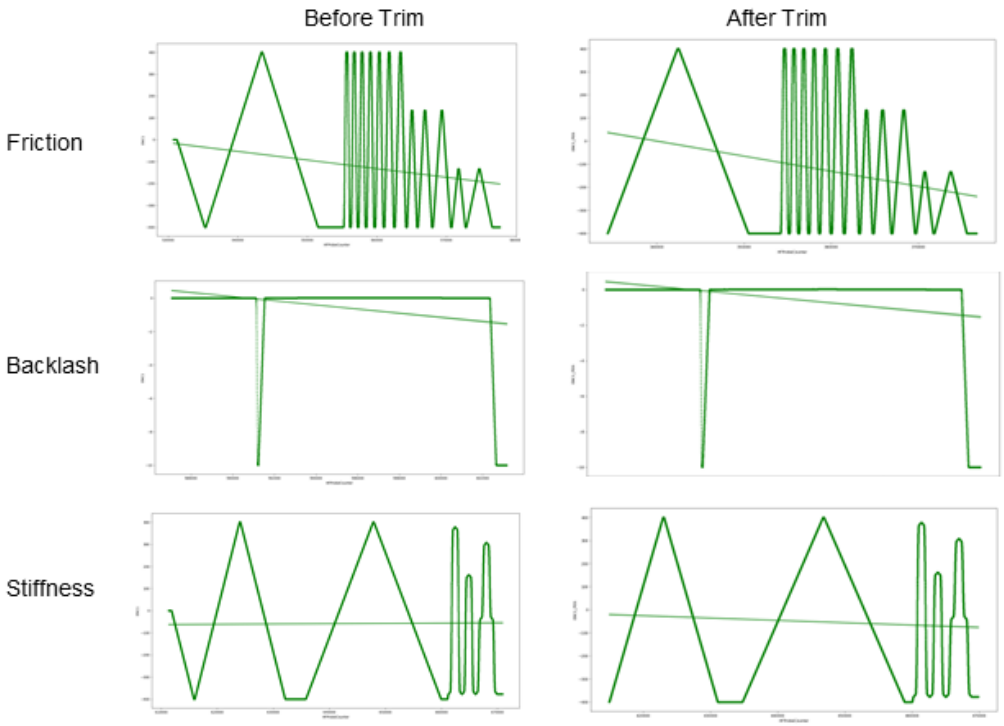

Figure 4.4: Regression plots of untrimmed and trimmed data for all the tests

Every test was considered separately for its feature development. Following are the results of the same:

- **Friction Test**: Figure 4.5 shows the input dataset developed for training the friction test model. The features "ENC1-DES" and "ENC2-DES" were used to classify the data. The dataset was classified into four parts considering positive and negative friction. The classes are shown in Table 4.1 .

	DESENC1	Torque	Class1
0	-0.00001	-1.524023	-L
1	-0.00001	-1.514062	-L
2	0.00000	-1.514062	+L
3	-0.00001	-1.514062	-L
4	-0.00001	-1.514062	-L
...	...	...	...
67781	0.00001	-1.703320	+L
67782	0.00002	-1.703320	+L
67783	0.00002	-1.703320	+L
67784	0.00001	-1.703320	+L
67785	0.00000	-1.703320	+L

Drive

	DESENC2	Torque	Class2
0	0.002591	-1.524023	+L
1	0.002591	-1.514062	+L
2	0.002591	-1.514062	+L
3	0.002591	-1.514062	+L
4	0.002591	-1.514062	+L
...	...	...	...
67781	0.000116	-1.703320	+L
67782	0.000116	-1.703320	+L
67783	0.000116	-1.703320	+L
67784	0.000116	-1.703320	+L
67785	0.000116	-1.703320	+L

Guideway

Figure 4.5: Input dataset for Friction test

Table 4.1: Classes in Friction test (all values are in mm)

ENC1-DES or ENC2-DES	Class
Below -25	-H
-25 to 0	-L
0 to 25	+L
Above 25	+H

Figure 4.6 shows the scatter plot of the data points in the final dataset. Since there is no faulty data, it is assumed that class "-H" and "+H" are the risk zones and any value in these zones must be considered as a potential machine calibration issue.

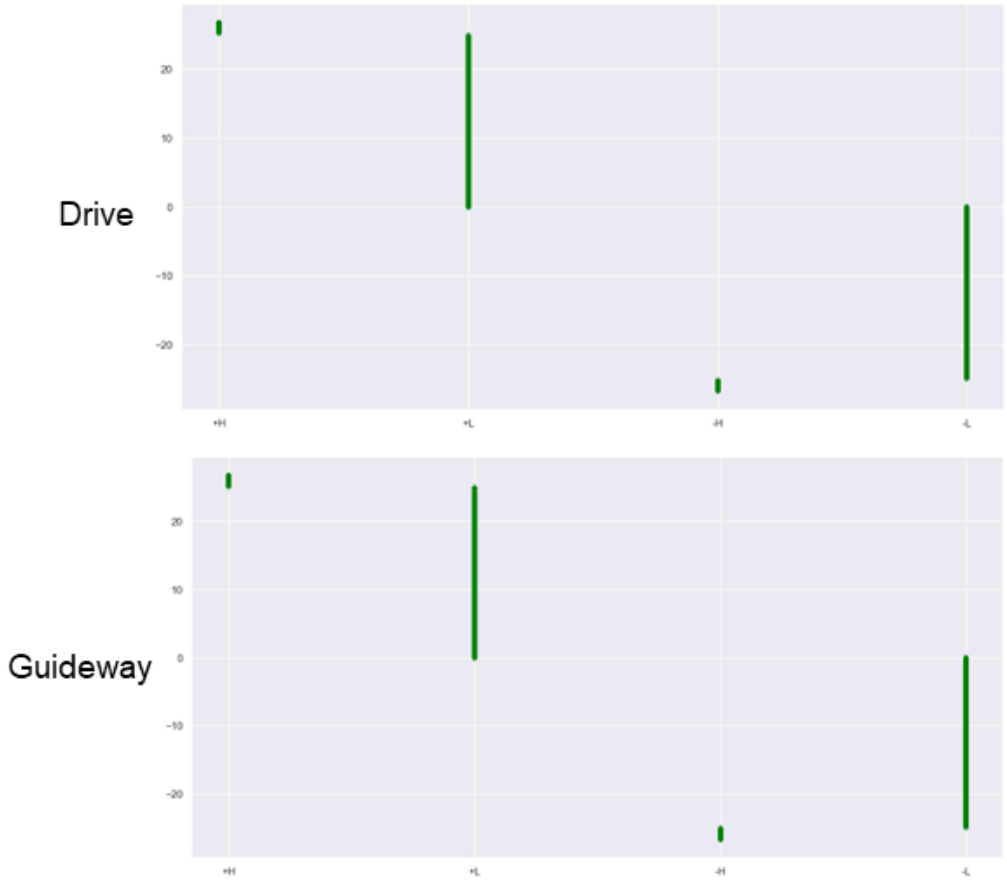

Figure 4.6: Scatter plot of Friction dataset

- **Backlash Test**: Figure 4.7 shows the input dataset developed for training the backlash test model. The feature "ENC2-ENC1" was used to classify the data. The dataset was classified into three parts considering the modulus value of ENC2-ENC1. The classes are shown in Table 4.2 .

	Diff	Class
0	0.004171	1.0
1	0.004183	1.0
2	0.004160	1.0
3	0.004171	1.0
4	0.004171	1.0
...	...	...
18949	0.001207	1.0
18950	0.001207	1.0
18951	0.001227	1.0
18952	0.001217	1.0
18953	0.001217	1.0

Figure 4.7: Input dataset for Backlash test

Table 4.2: Classes in Backlash test (all values are in mm)

ENC2-ENC1	Class
0 to 0.03	1
0.03 to 0.10	2
Above 0.10	3

Figure 4.8 shows the scatter plot of the data points in the final dataset. Since there is no faulty data, it is assumed that class "3" is the risk zone and any value in this zone must be considered as a potential machine calibration issue.

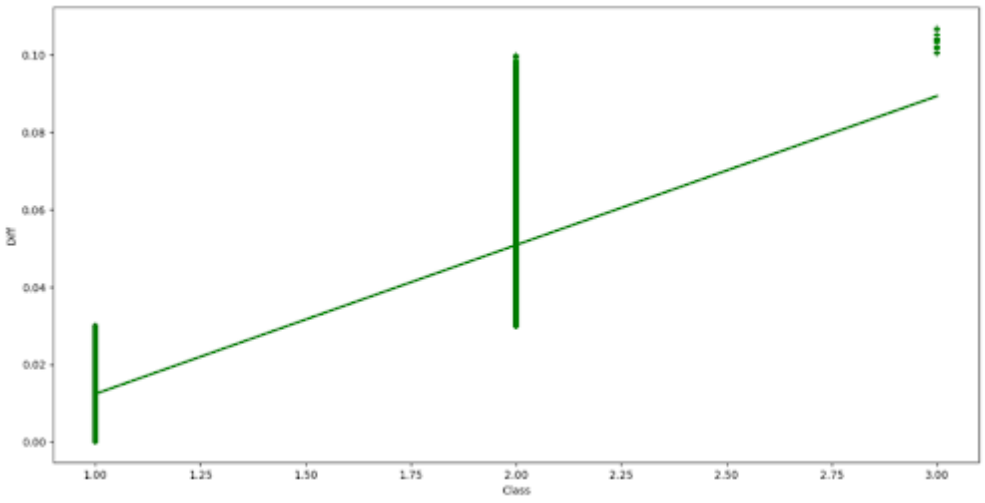

Figure 4.8: Scatter plot of Backlash dataset

- **Stiffness Test**: Figure 4.9 shows the input dataset developed for training the Stiffness test model. The feature "ENC2-ENC1" was used to classify the data. The dataset was classified into four parts considering increase and decrease in stiffness. The class to detect an increase in stiffness is "0" and the classes to

detect a decrease in stiffness are “1”, “2” and “3”. The classes are shown in Table 4.3 .

	Diff	Class
0	0.006839	1.0
1	0.006845	1.0
2	0.006849	1.0
3	0.006824	1.0
4	0.006867	1.0
...	...	...
66241	0.001823	1.0
66242	0.002025	1.0
66243	0.001789	1.0
66244	0.001369	1.0
66245	0.001156	1.0

Figure 4.9: Input dataset for Stiffness test

Table 4.3: Classes in Stiffness test (all values are in mm)

ENC2-ENC1	Class
Below 0	0
0 to 0.03	1
0.03 to 0.10	2
Above 0.10	3

Figure 4.10 shows the scatter plot of the data points in the final dataset. Since there is no faulty data, it is assumed that class “3” is the risk zone and any value in this zone must be considered as a potential machine calibration issue.

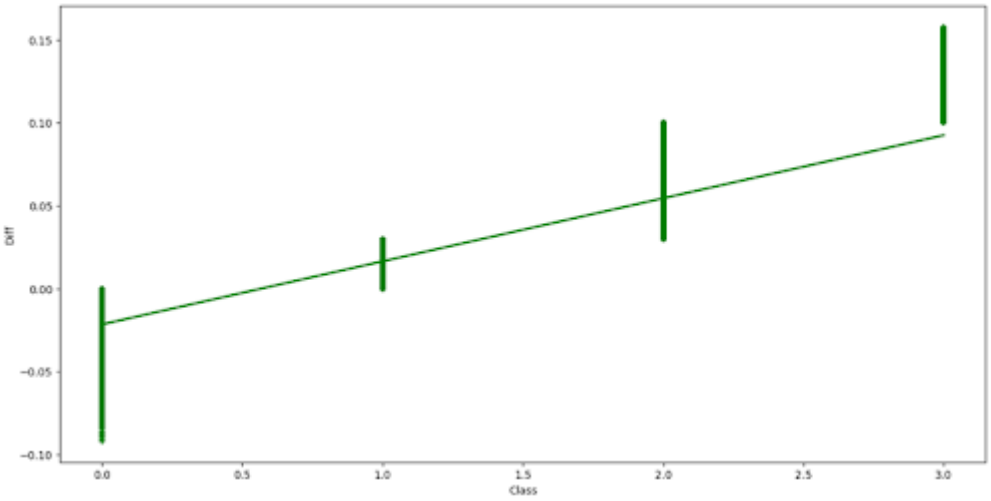

Figure 4.10: Scatter plot of Stiffness dataset

4.1.4 Modelling and Evaluation

The final dataset was from October 2021 to May 2022. This dataset was primarily split into two datasets, the first dataset consists of data from October 2021 to March 2022 and the second dataset (double validation data) consists of data from April 2022 to May 2022.

Three algorithms that were shortlisted for model preparation are "GNB", "DT" and "RF". The reason for selecting these algorithms was due to the fact that they are the most popular algorithms in multi-class classification models. As specified in section 2.5, a separate model was made for each test. The following sections depict the performance of each algorithm concerning various evaluation metrics considered for each test.

4.1.4.1 Friction Test

Two models were trained with two datasets, one for drive and one for the guide as explained in section 3.2.3.1. Table 4.4 shows the performance of the model when trained with different algorithms in terms of various evaluation metrics considered. All the models were trained with two variables as explained in section 3.2.3.1. With the help of table 4.4, the most suited algorithm was selected and test data was used to validate the model.

Table 4.4: Performance Metric comparison for Friction test

Algorithm	Accuracy	Precision	Recall	F1-Score
GNB – Drive	0.81948	0.84542	0.81948	0.81561
GNB – Guide	0.94355	0.94630	0.94355	0.94332
DT – Drive	1.00000	1.00000	1.00000	1.00000
DT – Guide	1.00000	1.00000	1.00000	1.00000
RF – Drive	1.00000	1.00000	1.00000	1.00000
RF – Guide	1.00000	1.00000	1.00000	1.00000

From Table 4.4 it is evident that both DT and RF algorithms performed equally good where the accuracy is 100%, precision is 100%, recall is 100% and F1-score is 100%. But it was concluded to go ahead with a model trained with the DT algorithm as it is simpler, comparatively faster and does not require rigorous training. In addition to that RF might be too powerful in this case as only two features are required to train the model. Figure 4.11 represents the Confusion matrix for the DT model on the training dataset.

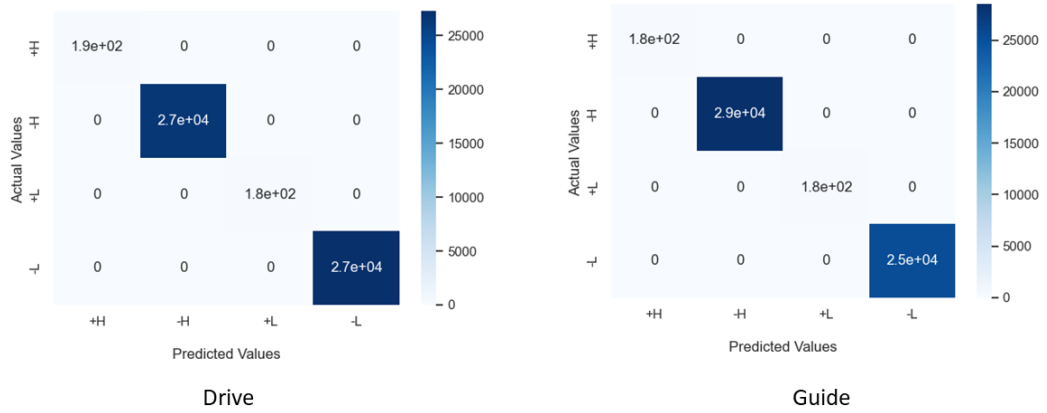

Figure 4.11: Confusion Matrix for Friction Test on Training dataset

The DT models were trained and the test dataset which was reserved initially (20% of the first dataset) for validation was used on these models to perform predictions. Figure 4.12 represents the Confusion matrix for the DT model on the test dataset. The validation resulted in an accuracy of 99.98%, the precision of 99.98%, a recall of 99.98% and an F1 score of 99.98% for the Drive and an accuracy of 99.99%, precision of 99.99%, recall of 99.99% and F1 score of 99.99% for the Guide.

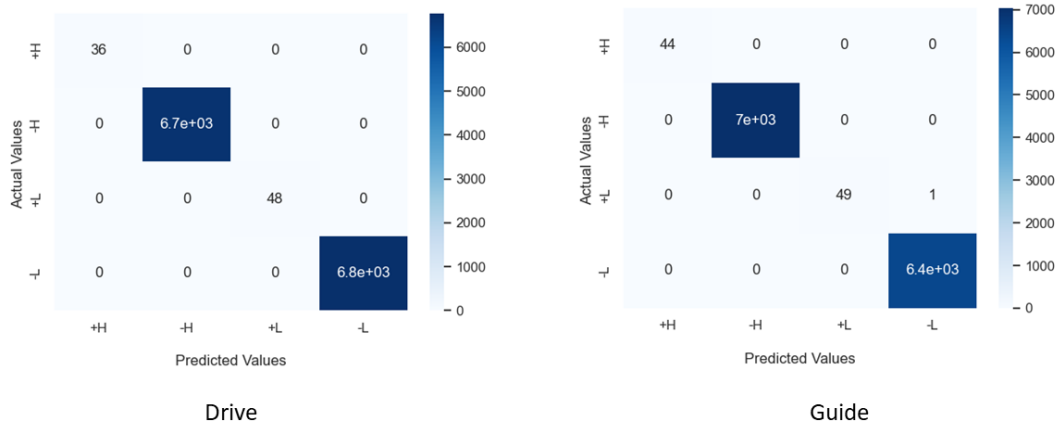

Figure 4.12: Confusion Matrix for Friction Test on Test dataset

For analyzing the performance of the model, double validation was also carried out. As specified above, the double validation data which is from April 2022 to May 2022 was classified through the rule-based method and was finally sent to the model for predictions. Figure 4.13 represents the Confusion matrix for the DT model on the double validation dataset. The double validation resulted in an accuracy of 99.99%, a precision of 99.99%, a recall of 99.99% and an F1-Score of 99.99% for both drive and guide.

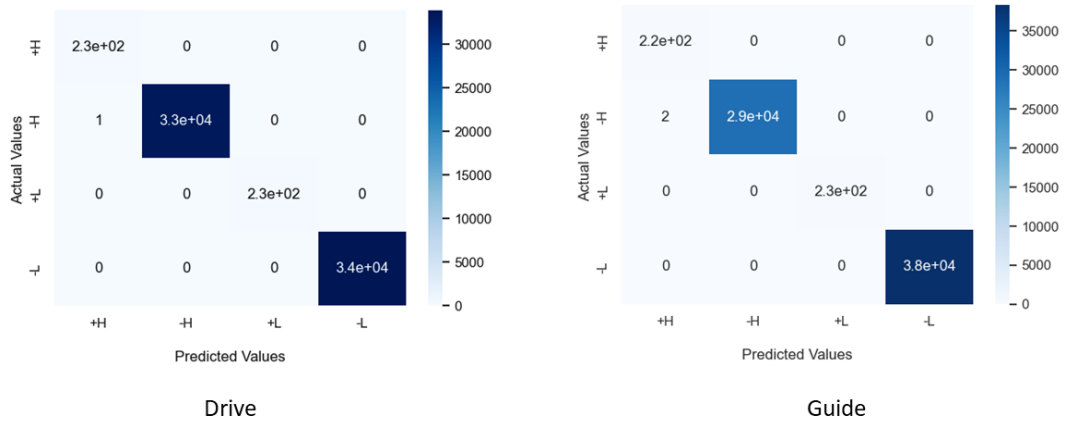

Figure 4.13: Confusion Matrix for Friction Test on Double validation dataset

When predicting the new data in the future, the model along with the predictions also displays the highest risk class values graphically. The company can view this graph and observe how many data points are in this high-risk class. This will help them in decision support of understanding the potential machine calibration issue till the model is re-trained with adequate faulty data. The graph that the model produces is depicted in Figure 4.14. The graph in Figure 4.14 is just an example and is not subjected to any test or double validation datasets used above.

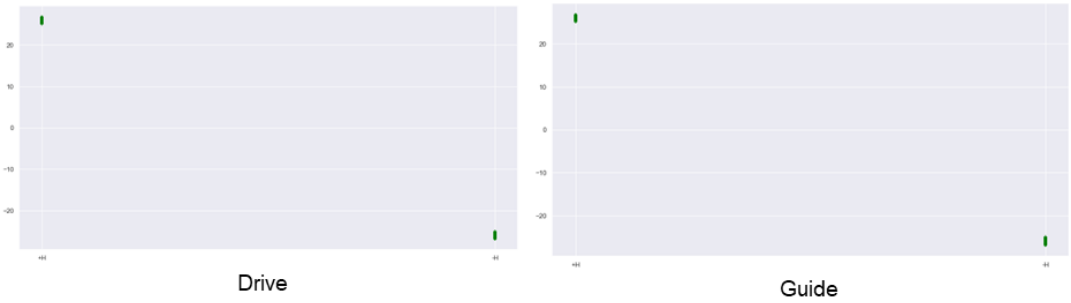

Figure 4.14: Representation of High risk class

4.1.4.2 Backlash Test

Table 4.5 shows the performance of the model when trained with different algorithms in terms of various evaluation metrics considered. All the models were trained with one feature, which was explained in section 3.2.3.1. With the help of Table 4.5, the most suited algorithm was selected and test data was run to validate the model.

Table 4.5: Performance Metric comparison for Backlash test

Algorithm	Accuracy	Precision	Recall	F1-Score
GNB	0.99220	0.99233	0.99220	0.99221
DT	0.99999	0.99999	0.99999	0.99999
RF	0.99999	0.99999	0.99999	0.99999

From Table 4.5 it is evident that both DT and RF algorithms performed equally good where the accuracy is 99.99%, precision is 99.99%, recall is 99.99% and F1-score is 99.99%. But it was concluded to go ahead with a model trained with the DT algorithm as it is simpler, comparatively faster and does not require rigorous training. In addition to that RF might be too powerful in this case as only one feature is required to train the model. Figure 4.15 represents the Confusion matrix for the DT model on the training dataset.

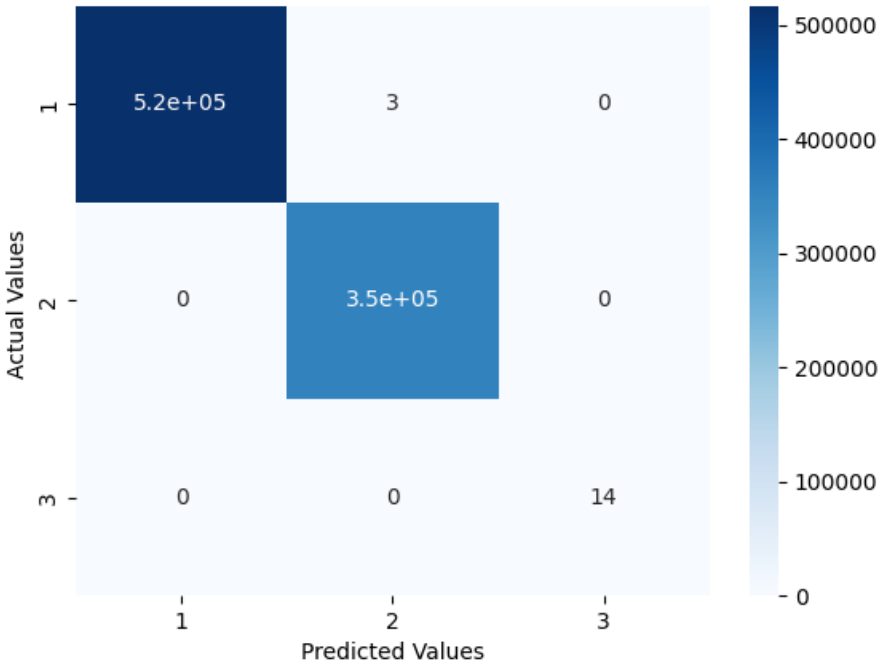

Figure 4.15: Confusion Matrix for Backlash Test on Training dataset

The DT model was trained and the test dataset which was reserved initially (20% of the first dataset) for validation was used on this model to perform predictions. Figure 4.16 represents the Confusion matrix for the DT model on the test dataset. The validation resulted in an accuracy of 99.99%, precision of 99.99%, recall of 99.99% and F1 score of 99.99%.

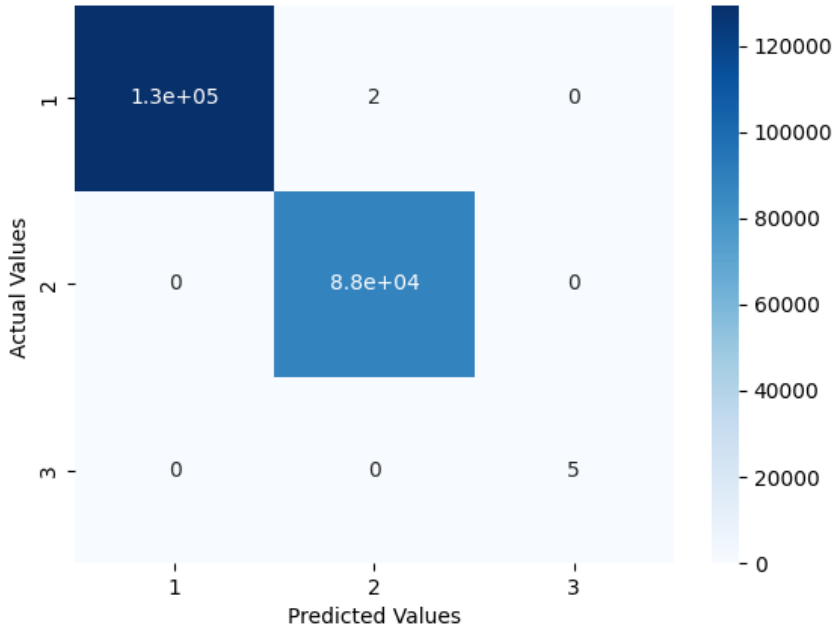

Figure 4.16: Confusion Matrix for Backlash Test on Test dataset

For analyzing the performance of the model, double validation was also carried out. As specified above, the double validation data which is from April 2022 to May 2022 was classified through the rule-based method and was finally sent to the model for predictions. Figure 4.17 represents the Confusion matrix for the DT model on the double validation dataset. The double validation resulted in an accuracy of 100%, a precision of 100%, a recall of 100% and an F1-Score of 100%.

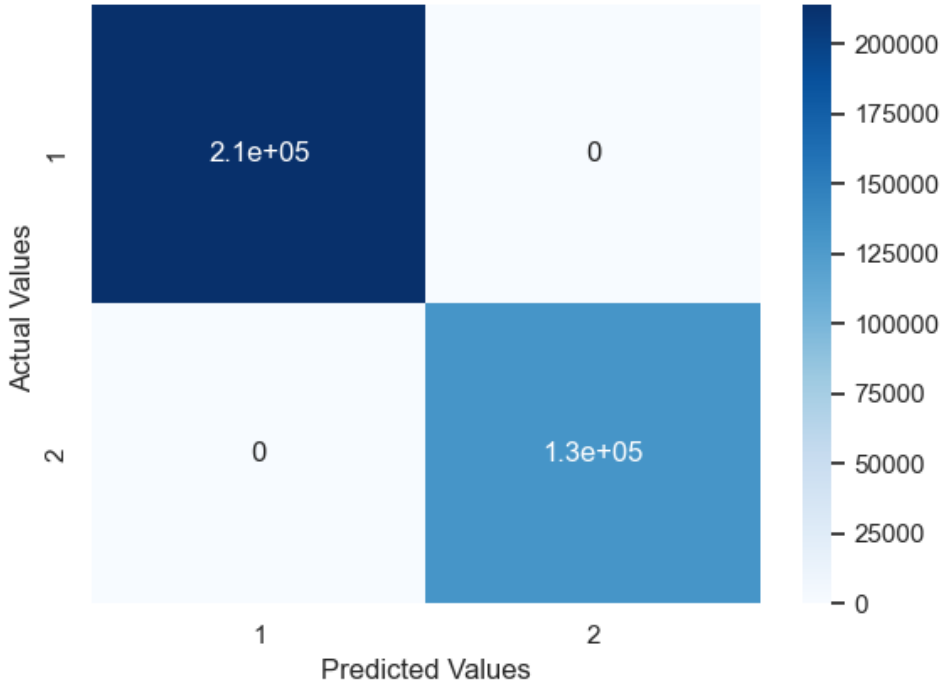

Figure 4.17: Confusion Matrix for Backlash Test on Double validation dataset

It should be noted that there was no data belonging to Class-3 in double validation dataset due to which Figure 4.17 does not have a representation of Class-3. Also, when predicting the new data in the future, the model along with the predictions also displays the highest risk class values graphically. The company can view this graph and observe how many data points are in this high-risk class. This will help them in decision support of understanding the potential machine calibration issue till the model is re-trained with adequate faulty data. The graph that the model produces is depicted in Figure 4.18. The graph in Figure 4.18 is just an example and is not subjected to any test or double validation datasets used above.

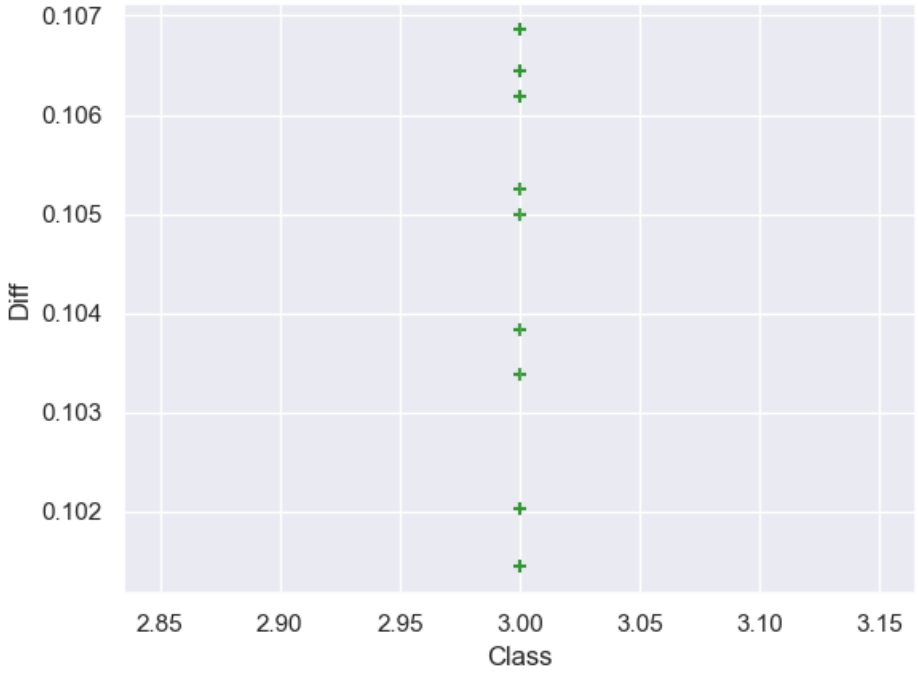

Figure 4.18: Representation of High risk class

4.1.4.3 Stiffness Test

Table 4.6 shows the performance of the model when trained with different algorithms in terms of various evaluation metrics considered. All the models were trained with one feature, which was explained in section 3.2.3.1. With the help of table 4.6, the most suited algorithm was selected and test data was run to validate the model.

Table 4.6: Performance Metric comparison for Stiffness test

Algorithm	Accuracy	Precision	Recall	F1-Score
GNB	0.97783	0.97840	0.97783	0.97783
DT	0.99999	0.99999	0.99999	0.99999
RF	0.99999	0.99999	0.99999	0.99999

From Table 4.6 it is evident that both DT and RF algorithms performed equally good where the accuracy is 99.99%, precision is 99.99%, recall is 99.99% and F1-score is 99.99%. But it was concluded to go ahead with a model trained with the DT algorithm as it is simpler, comparatively faster and does not require rigorous training. In addition to that RF might be too powerful in this case as only one feature is required to train the model. Figure 4.19 represents the Confusion matrix for the DT model on the training dataset.

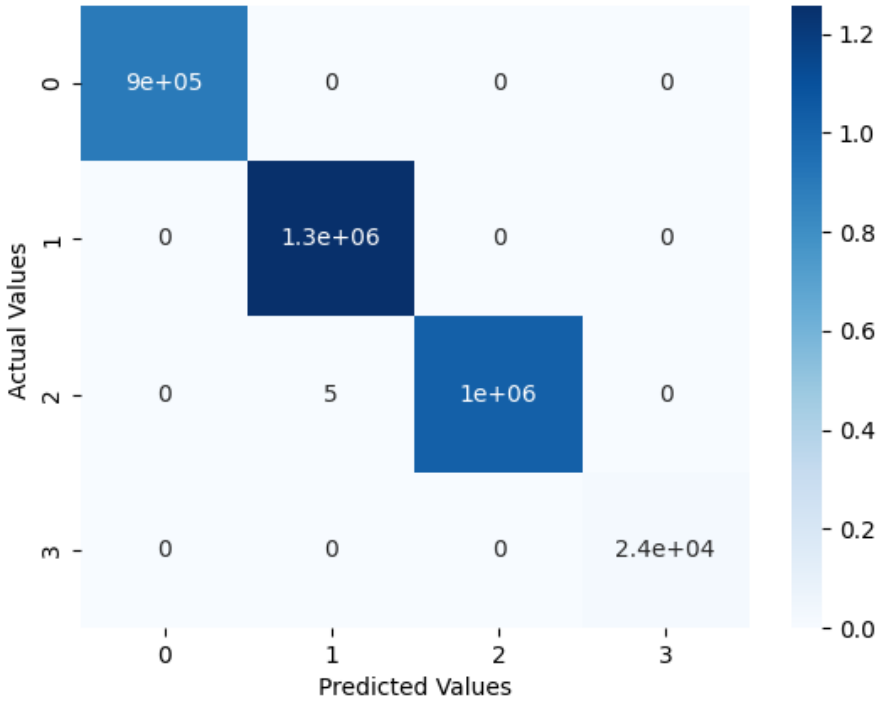

Figure 4.19: Confusion Matrix for Stiffness Test on Training dataset

The DT model was trained and the test dataset which was reserved initially (20% of the first dataset) for validation was used on this model to perform predictions. Figure 4.20 represents the Confusion matrix for the DT model on the test dataset. The preliminary validation resulted in an accuracy of 99.99%, precision of 99.99%, recall of 99.99% and F1 score of 99.99%.

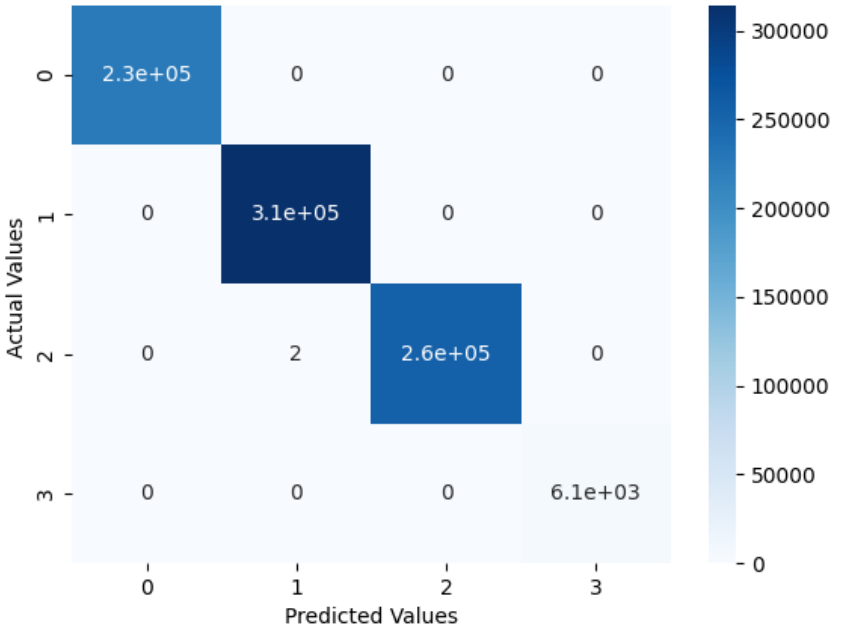

Figure 4.20: Confusion Matrix for Stiffness Test on Test dataset

For analyzing the performance of the model, double validation was also carried out. As specified above, the double validation data which is from April 2022 to May 2022 was classified through the rule-based method and was finally sent to the model for predictions. Figure 4.21 represents the Confusion matrix for the DT model on the double validation dataset. The double validation resulted in an accuracy of 100%, a precision of 100%, a recall of 100% and an F1-Score of 100%.

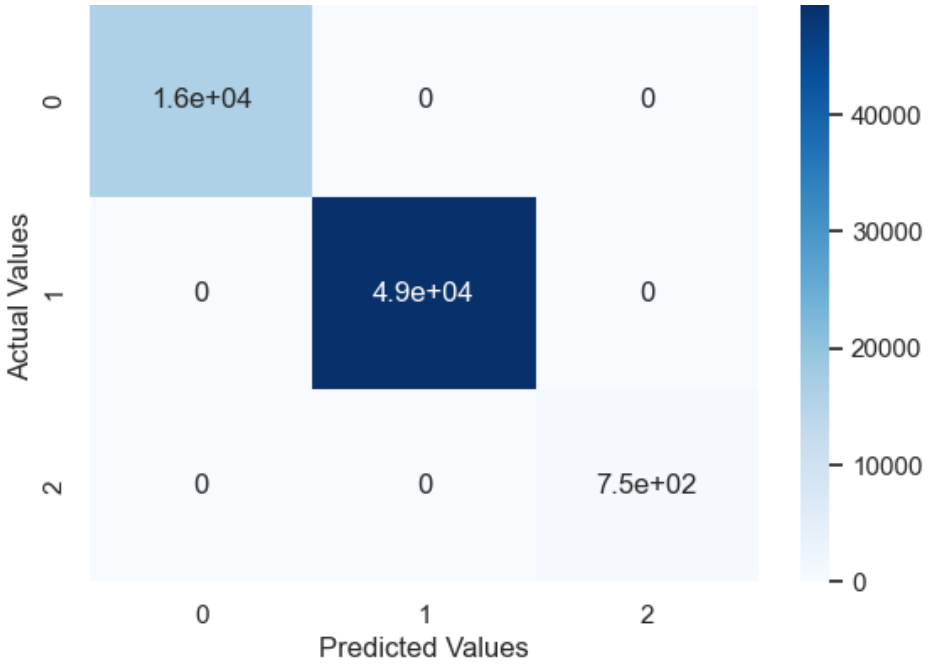

Figure 4.21: Confusion Matrix for Stiffness Test on Double validation dataset

It should be noted that there was no data belonging to Class-3 in double validation dataset due to which Figure 4.21 does not have a representation of Class-3. Also, when predicting the new data in the future, the model along with the predictions also displays the highest risk class values graphically. The company can view this graph and observe how many data points are in this high-risk class. This will help them in decision support of understanding the potential machine calibration issue till the model is re-trained with adequate faulty data. The graph that the model produces is depicted in Figure 4.22. The graph in Figure 4.22 is just an example and is not subjected to any test or double validation datasets used above.

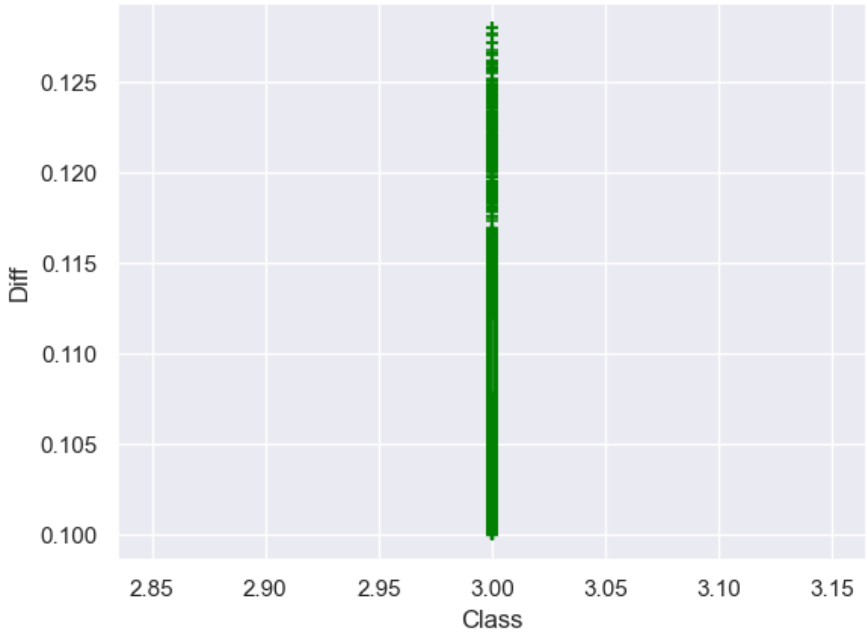

Figure 4.22: Representation of High risk class

4.2 Case Study 2: Manufacturing Company B

4.2.1 Business Understanding

The business objective of this case was understood by having meetings with involved stakeholders. Following are the stakeholders involved in this case:

- Manager Manufacturing IT/OT Systems of Manufacturing company B
- Reliability Engineer of Manufacturing company B
- Maintenance Engineer of Manufacturing company B
- PACA Project academic partners
- Academic Supervisors

After a few meetings with the above stakeholders, the business objective was defined. Currently, at Manufacturing company B, the raw data is being collected from the

Siemens Edge device, but it is not used at all. Therefore, Manufacturing company B wants to use this data for decision support in product quality checks of the parts being manufactured in the CNC machine. With the business objective being defined, a problem description was made which was explained in section 1.3.

4.2.2 Data Understanding

4.2.2.1 Data Quality Report

The data files and other supporting documents were provided by Manufacturing company B through a cloud platform called BOX which was very easy to access. But the data file formats are JSON files that are not so easily retrievable, unlike XML or TXT files. It took some time to retrieve the JSON files to ease the accessibility of the data. Along with data files, product quality data was provided in CSV format. The data was collected for every manufactured part during the production process and not in a standstill condition. All the data files were given in a single instance, as the data was already collected and stored by Manufacturing company B.

Specification of Data names and their definitions are given, for example, Encoders, Cycles, etc. The range of the values of the data files is missing which makes it difficult to comment if the given data exists in the range of known and accepted values. In addition to this, videos and photos were provided to understand the process and business rules. Data is considered credible as it comes directly from Manufacturing company B. Through meetings with the necessary stakeholders, it was observed that the data audits are carried out. A standard format is not followed amongst all the data files, the reason for the same was due to some experiments being carried out during the data retrieval part by the Reliability Engineer of Manufacturing company B. Some extra care was taken during the coding process to eliminate this issue.

After the pre-processing phase, the format of the data has been changed so that necessary features are used to train the ML model. As of now, the data is coming from only one source. It is also observed that data is consistent over time which was also verified by plotting the regression plot.For each manufactured part a unique Serial Number is being assigned. But in some instances, in Quality data and data files, due to some unknown reasons, the same serial number has been assigned to multiple manufactured parts. After discussing this with the corresponding stakeholders, it was decided to discard them during the data preparation phase. In addition, it was observed that some data files do not have the serial number, these files were ignored during the pre-processing phase of the ML model. It was interesting to see that the files with the missing serial number were on the same date and also around the same time.

The presence of domain knowledge will make the given data understandable to some extent. Therefore, it is not so easy to judge if the data meets the needs or not. The classification of data was easily understandable. In addition to that, it was easy to transform the semi-structured data into a structured format.

4.2.2.2 Exploratory Data Analysis

The data exploration stage led to certain insights about the data provided from the edge device. It was found that the headers in the files were not in the same order throughout the data collection period. The order of the headers in two different files is presented in Figure 4.23. To counter this situation, a general code was written which can extract the data even if the headers are not in sequential order.

```
File 1: ['TORQUE|1', 'TORQUE|7', 'TORQUE|14', 'ENC1_POS|1', 'ENC1_POS|7', 'ENC1_POS|14', 'DES_POS|1', 'DES_POS|7', 'DES_POS|14']
File 2: ['DES_POS|1', 'DES_POS|7', 'DES_POS|14', 'TORQUE|1', 'TORQUE|7', 'TORQUE|14', 'ENC1_POS|1', 'ENC1_POS|7', 'ENC1_POS|14']
```

Figure 4.23: Variation in the sequence of headers

The box plots of faulty and non-faulty data of the same variant was plotted as shown in Figure 4.24, to understand the variation in data points. It was observed that the maximum and minimum values are the same in both the cases but the data points within this limit are scattered over a larger range in case of a faulty part.

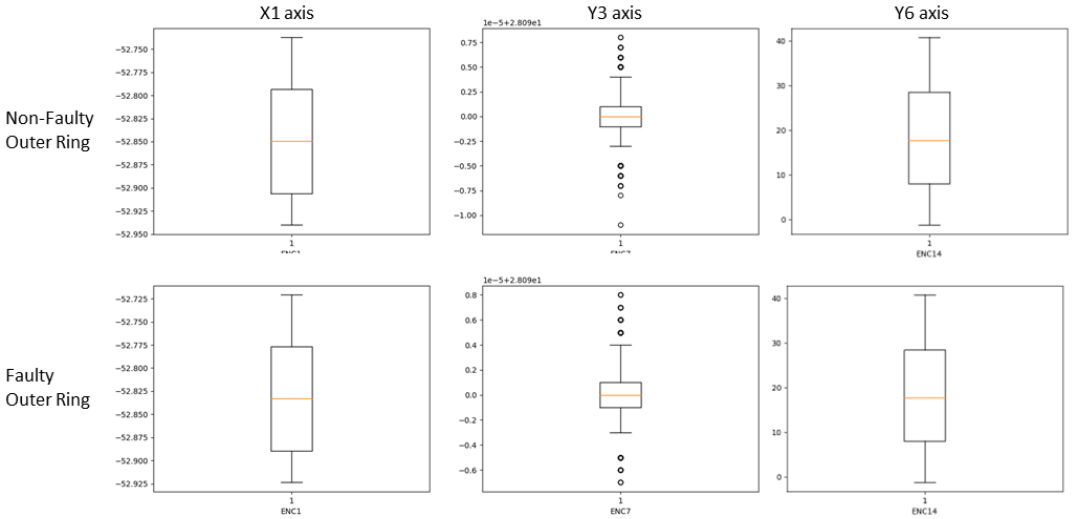

Figure 4.24: Box plots of non-faulty and faulty rings

Some inconsistencies were detected while merging the JSON files with the quality data. As mentioned in section 4.2.2.1, there were files with no serial numbers. The representation of the missing serial number is shown in Figure 4.25. Also, there were multiple instances where a part with the same serial number had multiple quality reports on the same data. Figure 4.26 shows an instance of a repetitive quality report of the same part. It was found that there are instances where JSON files with the same serial number on the same data appeared with different data points but there was only one quality report for its corresponding serial number and date. Figure 4.27 shows an instance of repetitive JSON files with only one quality report. These files corresponding to the above-stated issues were discarded after confirmation from the stakeholders.

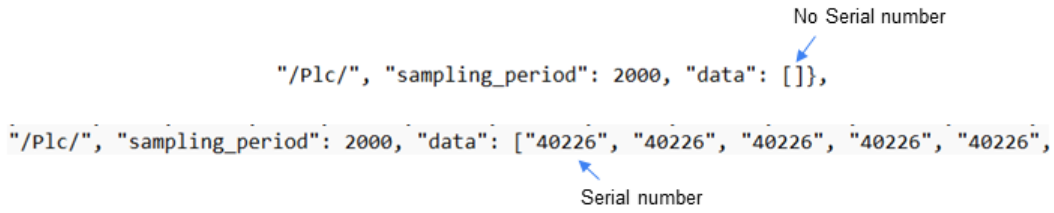

Figure 4.25: Missing Serial number

serial	timestamp	product	measur	toleran	toleran	out_of_tolerance
40353	13-04-2022 13:50	OR-23122 CC/C4W33	231,46	194.5	224.5	1
40353	13-04-2022 13:54	OR-23122 CC/C4W33	230,42	194.5	224.5	1

Figure 4.26: Repetitive Serial number in Quality report on same date

JSON Files

1		Minimum_ X1	Maximum _X1	Variance _X1	Minimum _Y3	Maximum _Y3	Variance _Y3	Minimum _Y6	Maximum _Y6	Variance _Y6	Sl_no	ime_Stamp
1224	629	-0.000253	-1.5E-05	-0.07203	0.000511	1.3E-05	0.072596	3.56E-09	8.255E-12	0.004939	40352	2022-04-13T11:20:28.205056Z
1225	1408	-0.000228	-1.1E-05	-0.07199	0.000557	1.1E-05	0.072612	4.61E-09	6.356E-12	0.004943	40352	2022-04-13T11:23:54.932639Z
1226	592	-0.000286	-1.4E-05	-0.07201	0.000569	1.4E-05	0.072571	5.9E-09	1.078E-11	0.004921	40352	2022-04-13T11:27:10.012854Z

Quality Report

serial	timestamp	product	measur	toleran	toleran	out_of_tolerance
40352	13-04-2022 13:40	OR-23122 CC/C4W33	599,99	194.5	224.5	1

Figure 4.27: Repetitive serial numbers with single Quality report on the same date

4.2.3 Data Preparation

All the JSON files were merged into a single dataframe and a feature extraction process was performed. The features extracted were minimum value, maximum value, mean, standard deviation, skewness and kurtosis. Peak value and Shape factor was not considered after analyzing the regression plots of the faulty and non-faulty data. It was observed that there were no defining peaks or changes in the shape of the graph. Figure 4.28 shows the comparison between axis movements during machining process of the faulty and non-faulty part.

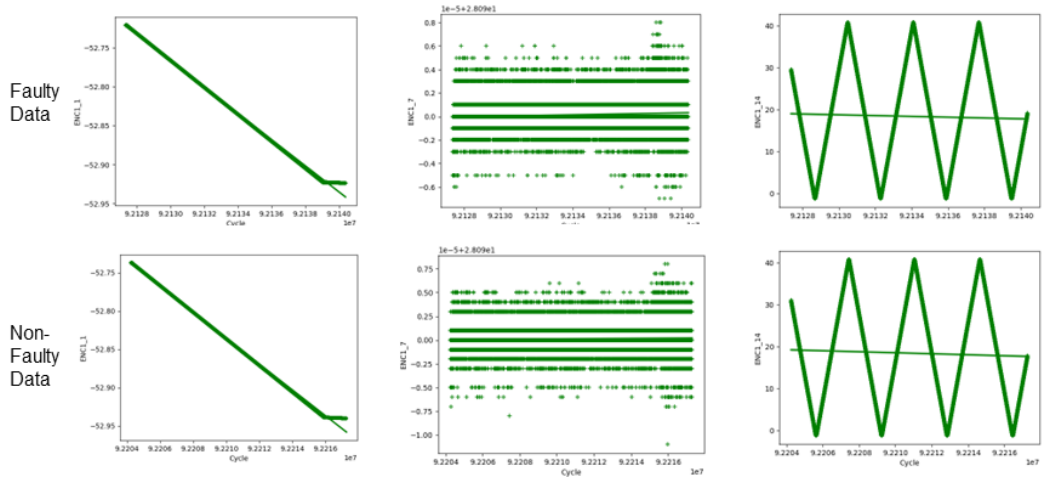

Figure 4.28: Regression plots of Faulty and Non-Faulty data

Further, quality data was merged to the feature table as per the serial numbers and the timestamps. Only the date in the timestamp was considered during the merging process as it was sufficient to match the files to its quality report. The resulting classes for the model were in line with the quality report of Manufacturing Company B which was explained in Section 3.2.3.2. Where Class '0' denotes the non-faulty part and Class '1' denotes the faulty part. Finally feature ranking was performed by using the F-score method and the top 10 and 15 features were considered for the modelling phase (Vakharia and Kankar, 2016). Figure 4.29 represents the feature ranking.

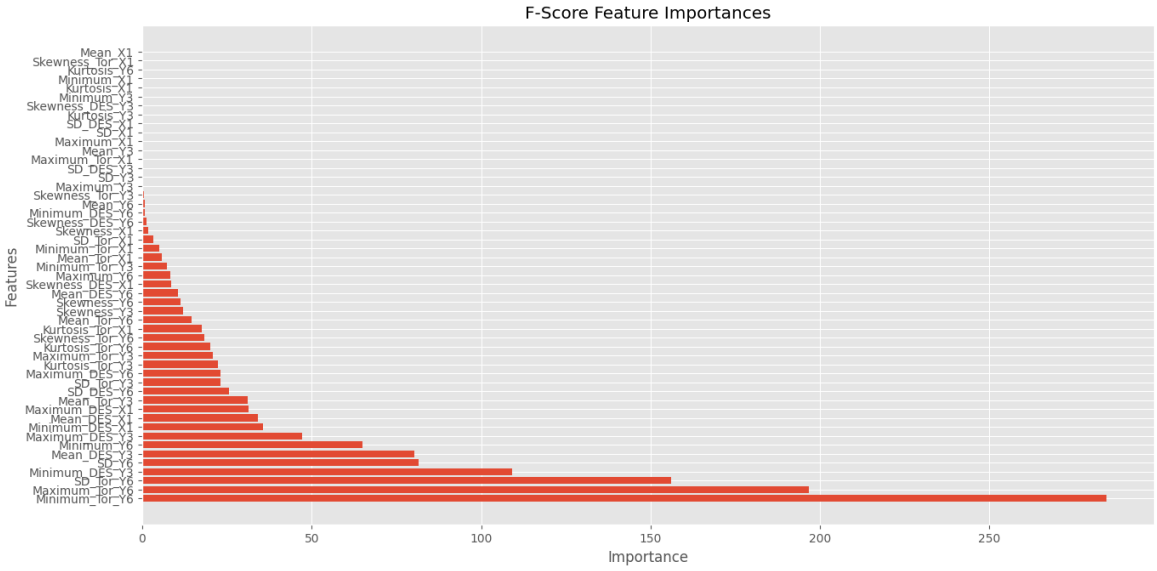

Figure 4.29: Feature ranking

After EDA, it was not expected to see the torque values having a major effect on the results but as per the ranking method, the torque values of the Y6 axis play a major

role in defining the model. Features pertaining to Y3 and Y6 axis are highly ranked and it was expected since these are the machining axes and X1 is a positioning axis.

4.2.4 Modelling and Evaluation

The final dataset was from 1st April 2022 to 13th May 2022. This dataset was primarily split into two datasets, the first dataset consists of data from 01st April 2022 to 30th April 2022 and the second dataset (double validation data) consists of data from 01st May 2022 to 13th May 2022. The final dataset of Manufacturing Company B was highly skewed where the fault data was around 3.18%. Therefore, the RUSBoost algorithm was chosen to train the model in this case as it is best suited for the imbalanced data and it follows the Random Under Sampling technique ("Classification with Imbalanced Data", n.d.). The models were trained with Top 10 and Top 15 features from the feature extraction analysis which was shown in section 4.2.3. In addition to that, the models were trained with several estimators of 30,50,100,150,200 and 250. Estimators in RUSBoost train several models internally where every model learns from the previous model and the final model learns all the hyper-parameters from the previous models. The number of estimators denotes the number of iterations within RUSBoost. Table 4.7 shows the performance of the model when trained with a different set of estimators and features in terms of various evaluation metrics considered. With the help of Table 4.7, the optimum number of estimators and features were selected and test data was run to validate the model.

Table 4.7: Performance Metric comparison

No.of Estimators	No.of Features	Accuracy	Precision	Recall	F1-Score
30	10	0.86325	0.96294	0.86325	0.90672
	15	0.90349	0.95749	0.90349	0.92816
50	10	0.78182	0.96884	0.78182	0.85589
	15	0.76976	0.95927	0.76976	0.84849
100	10	0.64174	0.95895	0.64174	0.76052
	15	0.81808	0.97136	0.81808	0.88072
150	10	0.77673	0.96045	0.77673	0.85254
	15	0.66136	0.96779	0.66136	0.77511
200	10	0.82263	0.96442	0.82263	0.88259
	15	0.82376	0.97117	0.82376	0.88399
250	10	0.78615	0.96668	0.78615	0.86026
	15	0.78132	0.96616	0.78132	0.85724

From Table 4.7, it can be inferred that the performance of the model is good when the number of estimators is 30 for both the top 10 and top 15 features. But, it was decided to not consider this as the optimum number of estimators since there would be a possibility that a certain set of hyper-parameter combinations would have been missed while training the model. From Table 4.7, the next best performing model is when the number of estimators is 200. The performance of the model when 200

estimators and top 10 features were selected has an accuracy of 82.26%, precision of 96.44%, recall of 82.26% and F1 score of 88.25%. Whereas, the performance of the model when 200 estimators and top 15 features were selected has an accuracy of 82.37%, precision of 97.11%, recall of 82.37% and F1 score of 88.39%. It is observed that there is a negligible difference in the performance of the model when the top 10 and top 15 features were trained. Therefore, considering the training time and data quantity to train the model, it is decided that the model with 200 estimators and top 10 features was the optimum value for the model. Figure 4.30 represents the Confusion matrix for the RUSBoost model on the training dataset.

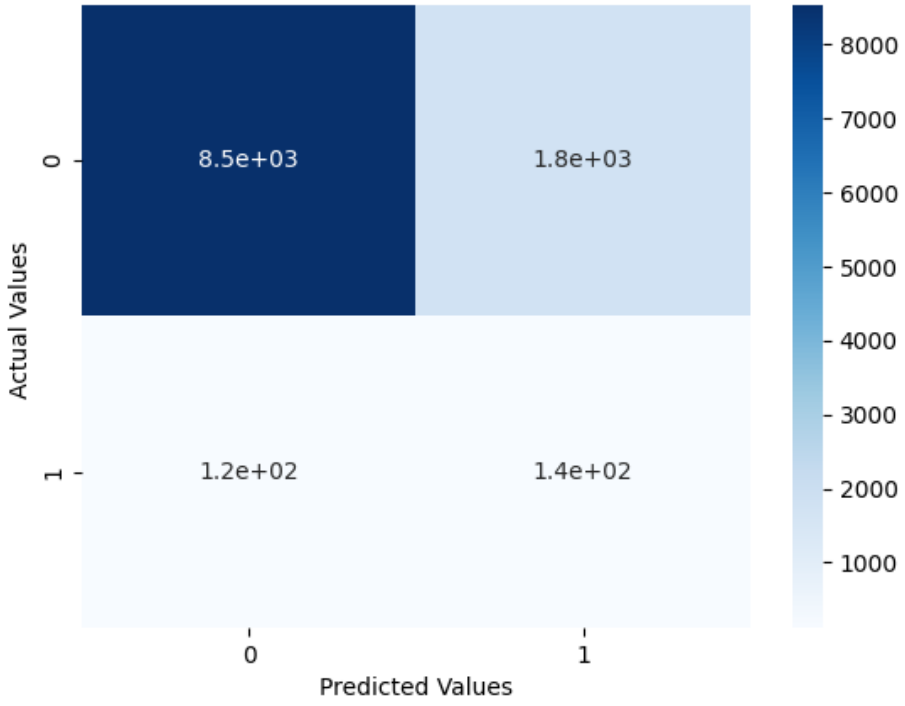

Figure 4.30: Confusion Matrix for RUSBoost model with 200 estimators and top 10 features on Training dataset

The model was trained and the test dataset which was reserved initially (20% of the first dataset) for validation was used on this model to perform predictions. Figure 4.31 represents the Confusion matrix for the model on the test dataset. The validation resulted in an accuracy of 83.25%, precision of 96.31%, recall of 83.25% and F1 score of 88.86%.

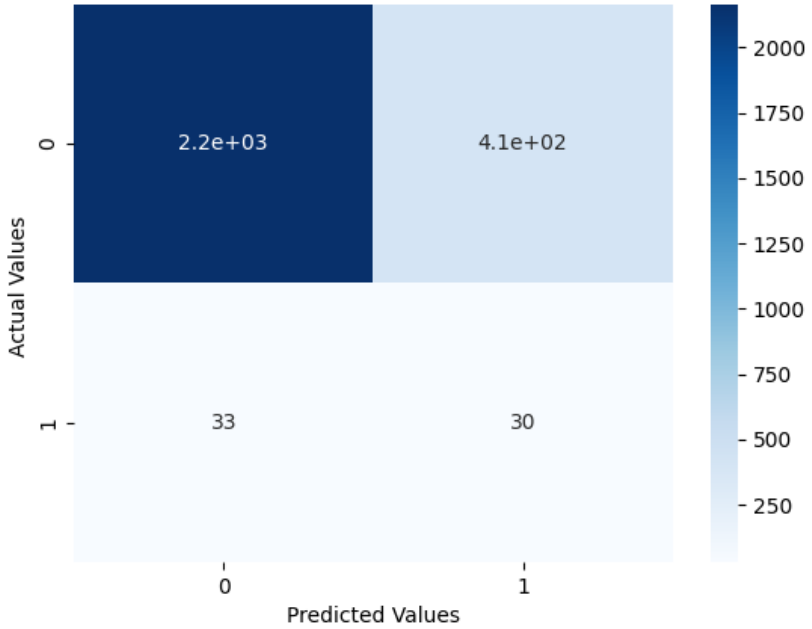

Figure 4.31: Confusion Matrix for RUSBoost model with 200 estimators and top 10 features on Test dataset

For analyzing the performance of the model, double validation was also carried out. As specified above, the double validation data which is from 1st May 2022 to 13th May 2022 was sent to the model for predictions. Figure 4.32 represents the Confusion matrix for the RUSBoost model on the double validation dataset. The double validation resulted in an accuracy of 93.57%, a precision of 96.72%, a recall of 93.57% and an F1-Score of 95.06%.

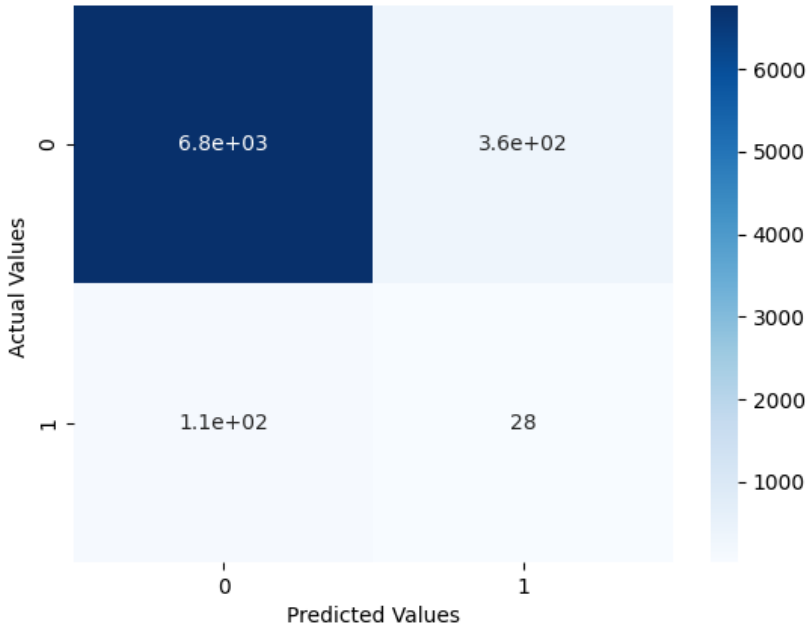

Figure 4.32: Confusion Matrix for RUSBoost model with 200 estimators and top 10 features on Double Validation dataset

5
Discussion

Processing data and providing real-time decision support are of key importance when it comes to the implementation of ML models in maintenance. Delay in decision support will lead to productivity losses due to downtime and abrupt stops in production, which nullifies the whole purpose of implementing an ML-based decision support system. As explained in section 2.4.1, the Edge device is one of the upcoming applications to provide real-time decision support. As an effort to answer the RQ1, a study was performed on the use of the Edge device in two different manufacturing setups. In both Manufacturing Company A and Company B, an Edge device was installed on the machines to capture high frequency data. This data was used to analyze the current condition of the machine through visualization tools. Since the data has raw values, simple formulations were developed to evaluate any variations in the machine. In the case of Manufacturing Company A, the formulations assumed were simple and specific to each test. This led to the development of a supervised ML model using the DT algorithm. The code developed for this case is general to any machine analyzing these specific tests using the same terminologies in the edge device. For Manufacturing Company B, the formulations were used to analyze the axis movements and further connect them to the quality report of each part. This led to a detailed analysis of the faulty and non-faulty parts. In this case, an ML model was developed using the RUSBoost algorithm to analyze the quality of the part immediately after the honing process.

The above-mentioned analysis and model development can be done using sensors as well, but the Edge device has its own advantages which can be used during the execution phase. As explained in section 2.4.1, a hybrid framework (3 layer framework) can be used in both cases. In the case of manufacturing Company A, a local cloud is present, which is linked to the Edge device. The developed ML model in this thesis can be deployed directly on the Edge device to provide a real-time calibration report. The re-training and further development of the ML model could be done in the local cloud. The data collected by the Edge device at any instance can be used to monitor the machine condition and further, the data can be sent to the local cloud to retrain the model. This would reduce the time gap between data collection and decision support which eventually can minimize the possibility of abrupt stops due to maintenance issues. Also, a filtration process can be done on the edge device to send only the unique data to the cloud to avoid retraining of the model on the same dataset. In addition to that, it would help them in reducing the communication requirements and burden on cloud services. It can also eliminate the security and privacy issues as data is locally processed on an edge device. Figure 5.1

shows the suggested framework for implementing Edge Intelligence in Manufacturing Company A.

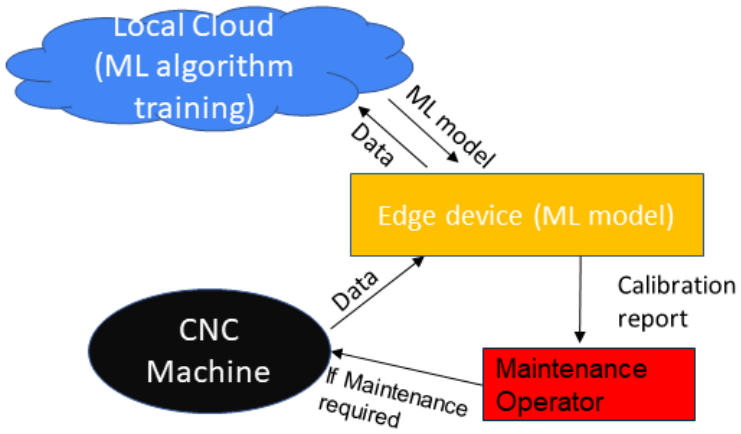

Figure 5.1: Framework for Edge Intelligence in Manufacturing Company A

In the case of manufacturing Company B, a local cloud is present, which is linked to the Edge device. The developed ML model in this thesis can be deployed directly on the Edge device to provide a real-time quality report. The re-training and further development of the ML model could be done in the local cloud. Deploying the model closer to the source would reduce the cycle time of the whole process. In addition to that, it would help them in reducing the communication requirements and burden on cloud services. It can also eliminate the security and privacy issues as data is locally processed on an edge device. Figure 5.2 shows the suggested framework for implementing Edge Intelligence in Manufacturing Company B.

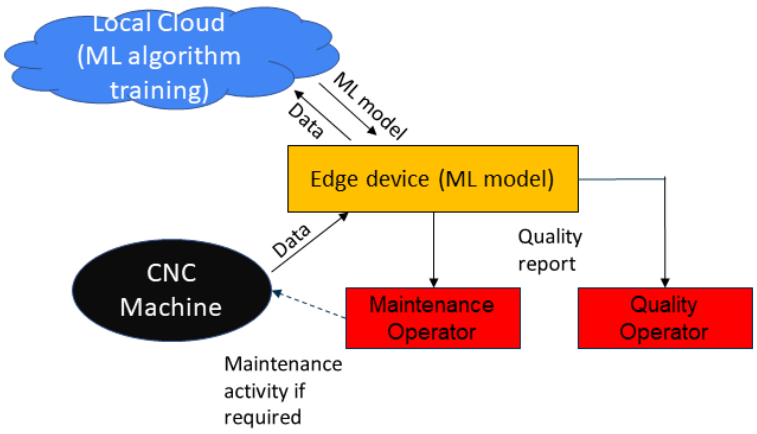

Figure 5.2: Framework for Edge Intelligence in Manufacturing Company B

For Manufacturing Company B, the faulty data was present which helped in the ease of analyzing the case and understanding the faulty conditions. This also helped in creating the formulations, extracting the necessary features and finally training the classification ML model. The precision and recall of the model is good, that means

it can predict the faulty and non faulty parts accurately upto 93.57%. The FP rate of the model is a bit high. This is due to the limited faulty data available to train the model.

To overcome this issue, another approach was taken where the RUSBoost algorithm was combined with a cross-validation technique. This was done to further reduce the effect of the biased dataset on the results. The model accuracy was similar to the above-described model but there was a trade-off between the FP and FN rates. The FP rate increased and the FN rate decreased in this model. Both the models will be provided to the company and the company can decide which model to choose depending on the point of application and the desired results from the model.

Retraining these models with more faulty data will eventually reduce the FP and FN rate and the predictions will be more accurate. During the whole process, there were no major assumptions made except to discard the data quality error files. Due to this, the model performance validation stands strong, the need to re-train the model is a few times and the handling of faulty data by the ML model can be clearly observed. Whereas for Manufacturing Company A, there was no faulty data as the CNC machine was very new. The absence of faulty data made it difficult to understand the faulty conditions of the machine. Therefore, assumptions were made with the formulations, variables and features considered for training the ML model. Due to the assumptions made, it is difficult to say if the model performance validation is strong enough and the number of times the model should be re-trained is comparatively higher. Also, currently, it cannot be observed how the ML model can handle the faulty data. So, it can be said that faulty data plays a crucial role in data preparation and modelling phases as it makes the model performance validation much stronger and reduces the assumptions during the above-mentioned phases. This addresses the RQ2 of this thesis.

After comparing both the case studies, it was seen that the implementation framework of Edge Intelligence is very similar to both the companies even though the processes are different. The analysis and outputs for Manufacturing Company A are very detailed and point out the root cause of failure but in the case of Manufacturing Company B, it only predicts the fault. This is due to the tests performed in the case of Manufacturing Company A, which help in analyzing the separate areas of the machine individually. Table 5.1 shows the comparison of both companies based on the presence of faulty data. It was seen that there were no assumptions required in the case of Manufacturing Company B as the faulty data was present, unlike Manufacturing Company A where certain assumptions were made. This led to fewer iterations of re-training the ML model and a reliable ML model in the case of Manufacturing Company B. Whereas, the ML model for Manufacturing Company A needs further development so that it can recognize faulty data when it comes into the picture. In simpler terms, the model developed for Manufacturing Company A is not completely reliable as it is not aware of the faulty data.

Table 5.1: Comparison of both companies based on the faulty data

Presence of faulty data (Manufacturing Company B)	Absence of faulty data (Manufacturing Company A)
No assumptions required.	Assumptions are required.
Fewer iterations to re-train the model.	Many iterations to re-train the model.
Model is aware of faulty conditions.	Model is unaware of faulty conditions.
More reliable for fault detection.	Less reliable for fault detection.

No research was carried out as the comparative study of ML models for PdM using edge device because edge device is a recent technology. Furthermore, there was not much research that was carried out in analyzing the calibration of the CNC machine in terms of tests like Friction, Backlash, Stiffness, Equability and Signature using the Edge device. As a result, this thesis can be considered as a foundation for further analysis on developing a robust ML model for calibration of a CNC machine using the tests stated before. In addition to that, few types of research were carried out in developing ML models for decision support in quality departments using Edge Intelligence. However, these ML models were supporting only a single variant product but the ML model developed in this thesis supports a multi-variant and multi-product.

5.1 Recommendations to Companies

This section presents the recommendations for both companies with some strategies.

5.1.1 Manufacturing Company A

The model developed for Manufacturing Company A is an iterative model due to the absence of faulty data. The risk levels present in the model are assumed and need to be further iterated to achieve true threshold values for axis calibration. To define the true threshold, it is recommended that the Company uses the quality report of the parts manufactured on the next day of the test. The error in tolerance of the parts can be traced back to the axis calibration tolerance. This variation in tolerance should be used as new risk levels in the model and retrain it. This iterative process should be continued until the model is completely trained on the basis of true threshold values.

5.1.2 Manufacturing Company B

As specified in section 1.3.2, Manufacturing Company B is collecting the axis movement data with the help of the Edge device during the honing process of the parts being manufactured. It would be recommended to deploy the ML model just after the honing process. Where the JSON files which are generated in real-time can be the input for the ML model and the model predicts the quality of the manufactured

part. This will save a lot of time and resources for the company in the quality inspection of the parts. It is advised to use this model as a complement to the existing quality inspection in the initial deployment phase. After the collection of more data, the model can be re-trained and can be used with its full potential decision support. The company can keep a tab on the number of faulty parts in a day and if there is a substantial increase in the faulty parts then they can check for the calibration of the CNC machine. If Manufacturing Company B wants to do a deep analysis on the calibration of the CNC machine, then they can even try to conduct the tests similar to Manufacturing Company A which was explained in section 2.4.2.

5.2 Contributions and Further Development

This thesis can be treated as one of the initial basements for the integration of ML-based decision support with Edge Intelligence for PdM. The Edge device is a new concept for both manufacturing companies. Therefore, this thesis can provide the companies with useful insights on how can they can further deploy it and get benefit from the applications of Edge Intelligence. This thesis also provides data quality evaluation as a part of the applied methodology CRISP-DM for the companies which they can work on the data extracted from the edge device further to improve and reduce the data quality issues. As specified earlier, the ML models developed through this thesis can save companies time and resources by facilitating the decision making process between the departments maintenance and quality. Through a comparative study, this thesis demonstrates the importance of faulty data in developing an ML model. Therefore, the results from this study can help understand the usability of ML algorithms in case of limited or missing faulty data in PdM applications.

This thesis findings were limited to developing the models. It can be interesting to see the model performance after the deployment, especially with Manufacturing company A with the presence of faulty data. During the thesis tenure, nothing was worked on the Edge device directly. There might be room for improvement in optimizing the data transfer among data source, edge device and cloud platform. For instance, the repetitive data can be filtered out as it is not so efficient to train the model with repetitive data. This can also ensure the reduction in data transfer and data storage.

For Manufacturing Company A, as specified in section 2.4, this thesis was carried out for only three tests due to the limitation of time and resources. In the future, an analysis can also be carried out for Equability and Signature tests. This can ensure Manufacturing Company A gets the results on all the possible tests for all the axes. Currently, for the Stiffness test, the steady state decrease in bearing stiffness is not considered. It can be interesting to see the change in formulations and model performance with its inclusion. At Manufacturing Company A, as specified in Section 1.3, currently a formulated analysis by Siemens is being used for evaluating the tests conducted by processing the raw data coming from the edge device. It can be useful to have a comparative study between Siemens formulated analysis and our ML models both in terms of accuracy and time efficiency.

For Manufacturing Company B, the dataset given was relatively small, which is around 30 days. Due to which the faulty data was also less, this might affect the performance of the developed model. Therefore, it would be interesting for the company to test and improved the model further with more data. In addition to that, the company can also incorporate the process parameters in the ML model building phase to make the ML model more robust and help them in scheduling maintenance activities.

6
Conclusion

This thesis provides a detailed exploratory study for PdM using Edge Intelligence by demonstrating in two different manufacturing cases. The first major point covered in this thesis is the implementation of the Edge device, an upcoming application for PdM activities to support the decision-making process. It has much more advantages than the sensors that are used to collect the data. The advantages like higher and faster processing capabilities and collection of raw data instead of signal values were studied. Considering these advantages, the strategy to deploy the ML model on the Edge device was made to provide real-time decision support. Another major point covered is the use of faulty data during the whole analysis. Faulty data provides great insights to track the machine condition and link it with the variation in the movement of the machine axis. It helps in data visualization by suggesting key features to be extracted for model development. Also, it was understood that faulty data plays a crucial role in the data preparation and modeling phase. The model developed using faulty data is much more reliable and accurate for decision support in real world implementation of PdM.

This thesis performs descriptive and predictive modelling by following a CRISP-DM methodology. The descriptive modelling was performed to explore the data in both the manufacturing cases and it resulted in understanding the machine condition, key variables in the data, presence of faulty data and formulations for the final dataset. Both the cases were analysed separately, where the data preparation relied mainly on the absence of faulty data in the case of Manufacturing Company A and the presence of faulty data in the case of Manufacturing Company B. Considering all the business goals, for Manufacturing Company A, a predictive model was developed to provide machine calibration details and for Manufacturing company B, a predictive model was developed to provide the quality report of the parts manufactured. Analyzing both cases helped in understanding the importance of faulty data to develop a reliable and accurate ML model.

The takeaway from this thesis is the framework developed for the implementation of Edge Intelligence which shows that the Edge device can reduce significant amounts of time delays as well as cost in a manufacturing setup. Further, by analyzing both the cases it can be said that, for the model to be fully functional it is essential to train the model on faulty data and define the classification classes. A detailed study of the machine, like in the case of Manufacturing Company A, helps in analyzing the root cause of failure and deciding the corresponding maintenance checks during maintenance activities. It is evident from the case of Manufacturing Company B

that the same model and setup can be used for scheduling maintenance activities along with the quality checks which will reduce the cost of developing two different setups to perform these activities.

This thesis successfully demonstrates the ML models developed for both cases and further provides a detailed implementation plan for the successful execution of the developed models with the help of the Edge device. This thesis can contribute to the use of Edge Intelligence for PdM activities with the extended knowledge from the analysis undertaken. Thereby, this thesis would also act as a stepping stone in the research for the complete usability of the Edge Intelligence in manufacturing.

Bibliography

Anti backlash for cnc. (n.d.). https://www.cnccookbook.com/anti-backlash-cnc-ball-screws-bearing-blocks-anti-backlash-nuts/

Bajaj, A. (2022). *Performance metrics in machine learning [complete guide]*. https://neptune.ai/blog/performance-metrics-in-machine-learning-complete-guide/

Cai, L., & Zhu, Y. (2015). The challenges of data quality and data quality assessment in the big data era. *Data Science Journal*, *14:2*, 1–10. https://datascience.codata.org/article/10.5334/dsj-2015-002/

Charbuty, B., & Abdulazeez, A. (2021). Classification based on decision tree algorithm for machine learning. *Journal of Applied Science and Technology Trends*, *2*(01), 20–28. https://jastt.org/index.php/jasttpath/article/view/65

Chaudhary, A., Kolhe, S., & Kamal, R. (2016). An improved random forest classifier for multi-class classification. *Information Processing in Agriculture*, *3*(4), 215–222. https://doi.org/https://doi.org/10.1016/j.inpa.2016.08.002

Çınar, Z. M., Abdussalam Nuhu, A., Zeeshan, Q., Korhan, O., Asmael, M., & Safaei, B. (2020). Machine learning in predictive maintenance towards sustainable smart manufacturing in industry 4.0. *Sustainability*, *12*(19). https://www.mdpi.com/2071-1050/12/19/8211

Classification with imbalanced data. (n.d.). https://www.mathworks.com/help/stats/classification-with-imbalanced-data.html

Dasu, T., & Johnson, T. (2003). *Exploratory data mining and data cleaning*. John Wiley & Sons, Inc.

Donges, N. (2022). *Random forest algorithm: A complete guide*. https://builtin.com/data-science/random-forest-algorithm

Draelos, R. (2019). *Measuring performance: The confusion matrix*. https://towardsdatascience.com/measuring-performance-the-confusion-matrix-25c17b78e516

Exploratory data analysis. (n.d.). https://www.jmp.com/en_hk/statistics-knowledge-portal/exploratory-data-analysis.html

Frank, A. G., Dalenogare, L. S., & Ayala, N. F. (2019). Industry 4.0 technologies: Implementation patterns in manufacturing companies. *International Journal of Production Economics*, *210*, 15–26. https://www.sciencedirect.com/science/article/pii/S0925527319300040

Gadre, M., & Deoskar, A. (2020). Industry 4.0 – digital transformation, challenges and benefits. *International Journal of Future Generation Communication and Networking*, *30*(2), 139–149. https://scholar.google.se/scholar?q=Industry+4.0+%E2%80%93+Digital+Transformation,+Challenges+and+Benefits&hl=en&as_sdt=0&as_vis=1&oi=scholart

Hafeez, T., Xu, L., & Mcardle, G. (2021). Edge intelligence for data handling and predictive maintenance in iiot. *IEEE Access*, *9*, 49355–49371. https://doi.org/10.1109/ACCESS.2021.3069137

Jahromi, A. H., & Taheri, M. (2017). A non-parametric mixture of gaussian naive bayes classifiers based on local independent features. *2017 Artificial Intelligence and Signal Processing Conference (AISP)*, 209–212. https://doi.org/10.1109/AISP.2017.8324083

Kaur, S., & Jindal, S. (2016). A survey on machine learning algorithms. *IJIRAE-International Journal of Innovative Research in Advanced Engineering*, *3*. https://www.ijirae.com/volumes/Vol3/iss11/02.NVAE10084.pdf

Kehayov, M., Holder, L., & Koch, V. (2022). Application of artificial intelligence technology in the manufacturing process and purchasing and supply management [3rd International Conference on Industry 4.0 and Smart Manufacturing]. *Procedia Computer Science*, *200*, 1209–1217. https://www.sciencedirect.com/science/article/pii/S1877050922003301

Kubiak, K., Dec, G., & Stadnicka, D. (2022). Possible applications of edge computing in the manufacturing industrymdash;systematic literature review. *Sensors*, *22*(7). https://doi.org/https://doi.org/10.3390/s22072445

Kumar, J., Soni, V., & Agnihotri, G. (2013). Maintenance performance metrics for manufacturing industry. *IJRET: International Journal of Research in Engineering and Technology*, *2*, 136–142. https://scholar.google.se/scholar?q=maintenance+performance+metrics+for+manufacturing+industry&hl=en&as_sdt=0&as_vis=1&oi=scholart

Lampropoulos, G., Siakas, K., & Anastasiadis, T. (2019). Internet of things in the context of industry 4.0: An overview. *International Journal of Entrepreneurial Knowledge*, *7*, 4–19. https://doi.org/10.2478/ijek-2019-0001

Lasi, H., Fettke, P., Kemper, H.-G., Feld, T., & Hoffmann, M. (2014). Industry 4.0. *Business Information Systems Engineering*, *6*, 239–242. https://doi.org/https://doi.org/10.1007/s12599-014-0334-4

Lee, J., Davari, H., Singh, J., & Pandhare, V. (2018). Industrial artificial intelligence for industry 4.0-based manufacturing systems. *Manufacturing Letters*, *18*, 20–23. https://doi.org/https://doi.org/10.1016/j.mfglet.2018.09.002

Lee, M.-X., Lee, Y.-C., & Chou, C. J. (2017). Essential implications of the digital transformation in industry 4.0. *Journal of Scientific and Industrial Research*, *76*, 465–467. http://nopr.niscair.res.in/handle/123456789/42548

Lené, J., & Rajashekarappa, M. (2021). *A data-driven approach to detect air leakage in a pneumatic system* (Master's thesis). Chalmers tekniska högskola / Institutionen för industri- och materialvetenskap. https://hdl.handle.net/20.500.12380/302631

Lundgren, C., Bokrantz, J., & Skoogh, A. (2021). A strategy development process for smart maintenance implementation. *Journal of Manufacturing Technology Management*, *32*, 142–166. https://doi.org/https://doi.org/10.1108/JMTM-06-2020-0222

Majumder, P. (n.d.). *Gaussian naive bayes*. https://iq.opengenus.org/gaussian-naive-bayes/

Marr, B. (2018). *What is industry 4.0? here's a super easy explanation for anyone*. https://www.forbes.com/sites/bernardmarr/2018/09/02/what-is-industry-4-0-heres-a-super-easy-explanation-for-anyone/?sh=60ccdaff9788

Montero Jimenez, J. J., Schwartz, S., Vingerhoeds, R., Grabot, B., & Salaün, M. (2020). Towards multi-model approaches to predictive maintenance: A systematic literature survey on diagnostics and prognostics. *Journal of Manufacturing Systems*, *56*, 539–557. https://www.sciencedirect.com/science/article/pii/S0278612520301187

Narkhede, S. (2018). *Understanding confusion matrix*. https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62

Patel, H. H., & Prajapati, P. (2018). Study and analysis of decision tree based classification algorithms. *International Journal of Computer Sciences and Engineering*, *6*, 74–78. https://www.ijcseonline.org/full_paper_view.php?paper_id=2984

Pelt, Maurice, Stamoulis, Konstantinos, & Apostolidis, Asteris. (2019). Data analytics case studies in the maintenance, repair and overhaul (mro) industry. *MATEC Web Conf.*, *304*, 04005. https://doi.org/10.1051/matecconf/201930404005

Qi, Q., & Tao, F. (2019). A smart manufacturing service system based on edge computing, fog computing, and cloud computing. *IEEE Access*, *7*, 86769–86777. https://doi.org/10.1109/ACCESS.2019.2923610

Ran, Y., Zhou, X., Lin, P., Wen, Y., & Deng, R. (2019). A survey of predictive maintenance: Systems, purposes and approaches. https://arxiv.org/abs/1912.07383

Seiffert, C., Khoshgoftaar, T. M., Van Hulse, J., & Napolitano, A. (2008). Rusboost: Improving classification performance when training data is skewed. *2008 19th International Conference on Pattern Recognition*, 1–4. https://doi.org/10.1109/ICPR.2008.4761297

Susto, G. A., Schirru, A., Pampuri, S., McLoone, S., & Beghi, A. (2015). Machine learning for predictive maintenance: A multiple classifier approach. *IEEE Transactions on Industrial Informatics*, *11*(3), 812–820. https://doi.org/10.1109/TII.2014.2349359

Tanha, J., Abdi, Y., Samadi, N., Razzaghi, N., & Asadpour, M. (2020). Boosting methods for multi-class imbalanced data classification: An experimental review. *Journal of Big data*, *7*, 70. https://doi.org/https://doi.org/10.1186/s40537-020-00349-y

Tavish. (2019). *11 important model evaluation metrics for machine learning everyone should know*. https://www.analyticsvidhya.com/blog/2019/08/11-important-model-evaluation-error-metrics/

Todorovac, K., & Wiking, N. (2021). An exploratory study of manufacturing data and its potential for continuous process improvements from a production economical perspective. https://scholar.google.se/scholar?q=An+exploratory+study+of+manufacturing+data+and+its+potential+for+continuous+process+improvements+from+a+production+economical+perspective&hl=en&as_sdt=0&as_vis=1&oi=scholart

Vakharia, V., & Kankar, V. K. G. P. K. (2016). A comparison of feature ranking techniques for fault diagnosis of ball bearing. *Soft Comput*, *20*, 1601–1619. https://doi.org/https://doi.org/10.1007/s00500-015-1608-6

Vasudevan, A., & Duan, X. (2021). *A systematic data science approach towards predictive maintenance application in manufacturing industry* (Master's thesis). Chalmers tekniska högskola / Institutionen för industri- och materialvetenskap. https://hdl.handle.net/20.500.12380/302632

Wickham, H. (2014). Tidy data. *Journal of Statistical Software*, *59*(10), 1–23. https://doi.org/10.18637/jss.v059.i10

Wirth, R., & Hipp, J. (2000). Crisp-dm: Towards a standard process model for data mining. *Proceedings of the 4th International Conference on the Practical Applications of Knowledge Discovery and Data Mining*. https://www.researchgate.net/publication/239585378_CRISP-DM_Towards_a_standard_process_model_for_data_mining

Yang, T., Yi, X., Lu, S., Johansson, K. H., & Chai, T. (2021). Intelligent manufacturing for the process industry driven by industrial artificial intelligence. *Engineering*, *7*(9), 1224–1230. https://www.sciencedirect.com/science/article/pii/S2095809921003064

Zeba, G., Dabić, M., Čičak, M., Daim, T., & Yalcin, H. (2021). Technology mining: Artificial intelligence in manufacturing. *Technological Forecasting and Social Change*, *171*, 120971. https://www.sciencedirect.com/science/article/pii/S0040162521004030

A

Exploratory Data Analysis A

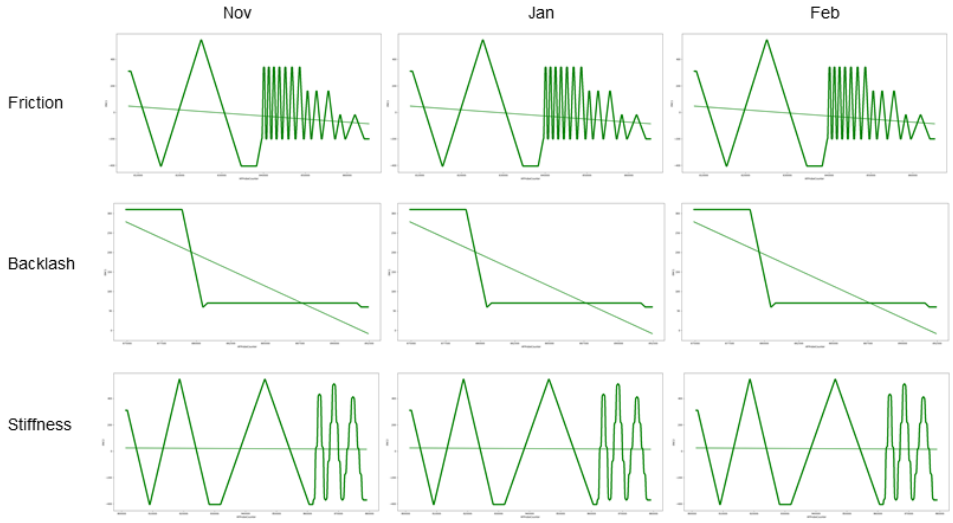

Figure A.1: Regression Plots of Y-axis of the tests over a period of time

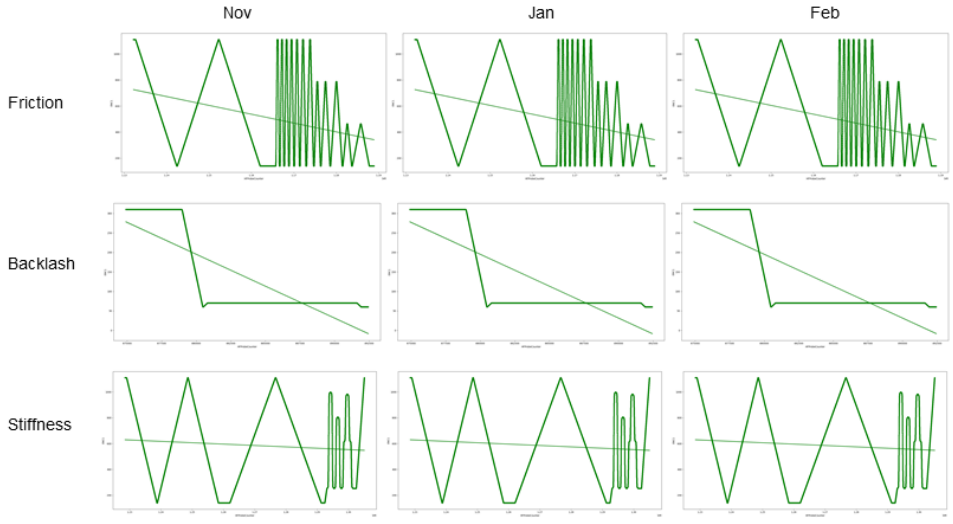

Figure A.2: Regression Plots of Z-axis of the tests over a period of time

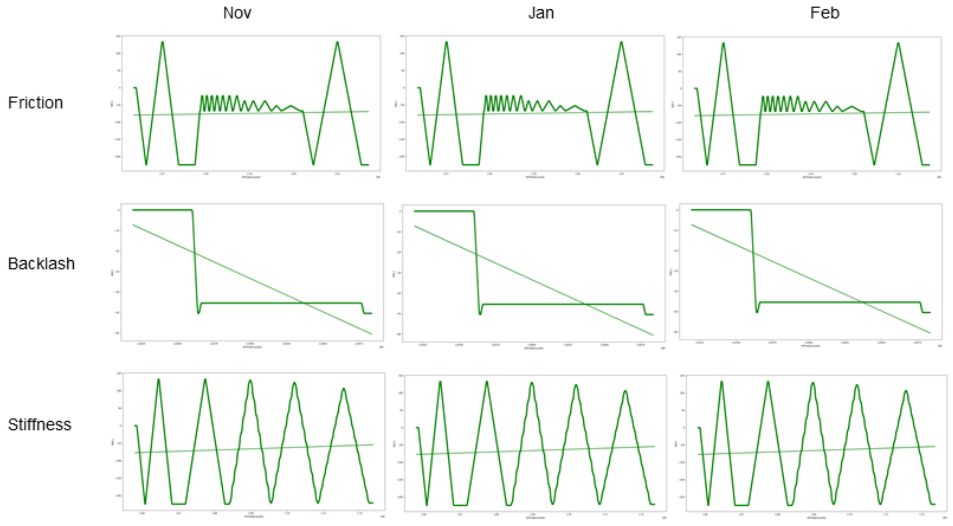

Figure A.3: Regression Plots of A-axis of the tests over a period of time

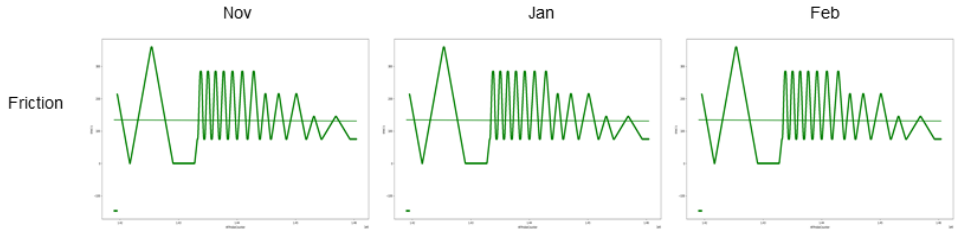

Figure A.4: Regression Plots of B-axis of the tests over a period of time

B

Exploratory Data Analysis B

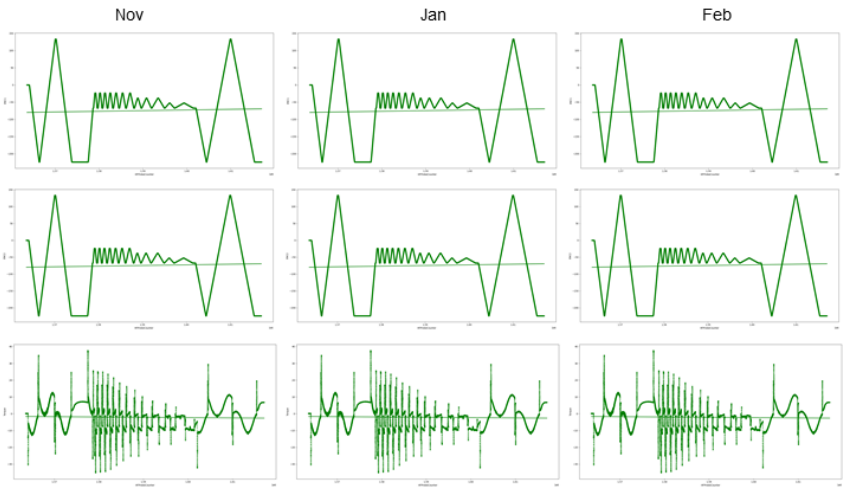

Figure B.1: Encoder and Torque comparison for friction test over a period of time of A-axis

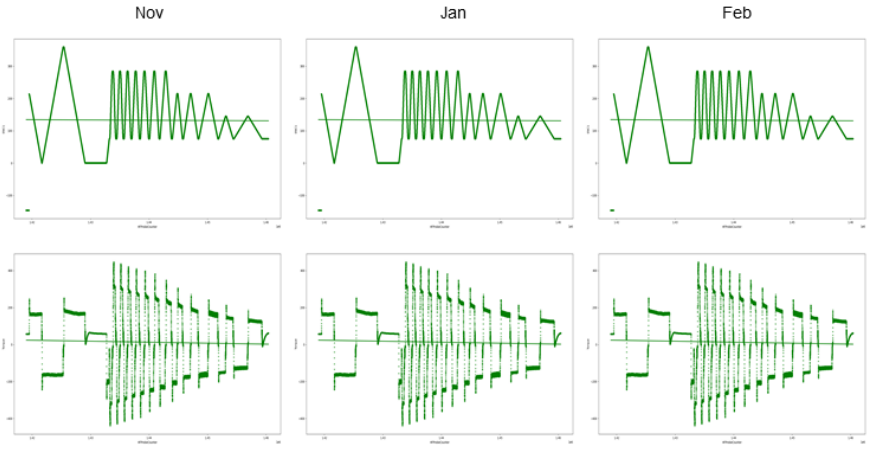

Figure B.2: Encoder and Torque comparison for friction test over a period of time of B-axis

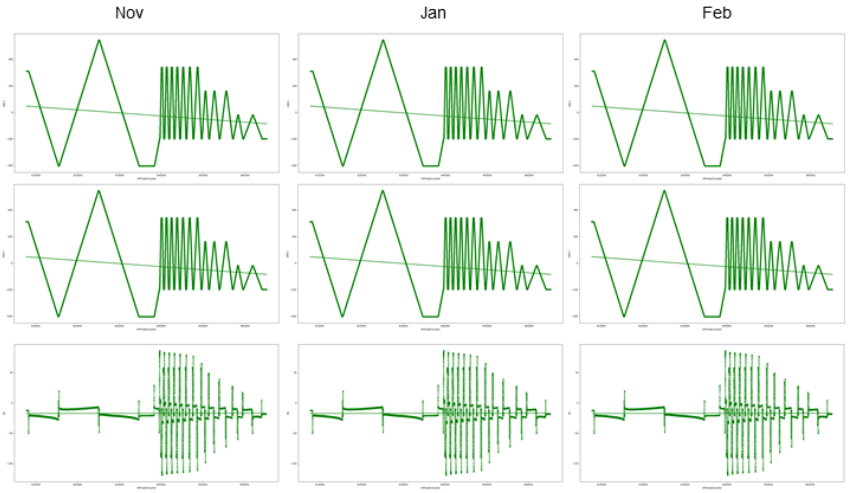

Figure B.3: Encoder and Torque comparison for friction test over a period of time of Y-axis

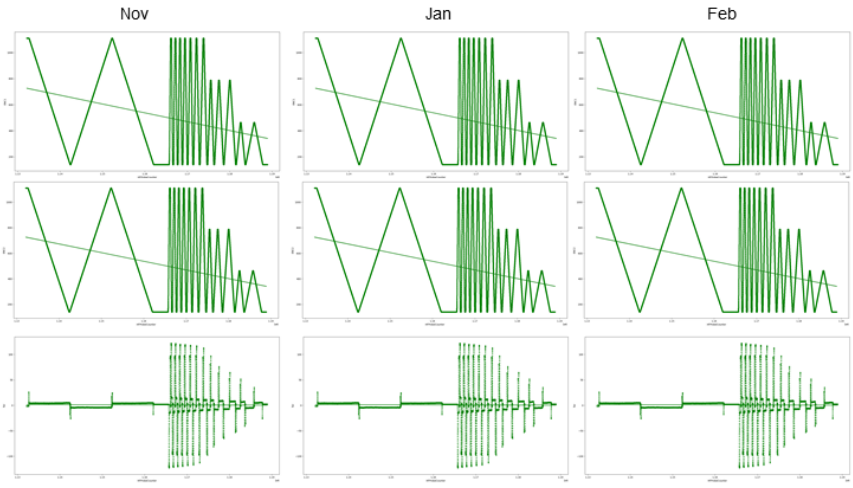

Figure B.4: Encoder and Torque comparison for friction test over a period of time of Z-axis

DEPARTMENT OF INDUSTRIAL AND MATERIALS SCIENCE
CHALMERS UNIVERSITY OF TECHNOLOGY
Gothenburg, Sweden
www.chalmers.se

CHALMERS
UNIVERSITY OF TECHNOLOGY